(2)

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

SEARCH AND RESCUE AND ENFORCEMENT OF LAWS AND TREATIES INTELLIGENCE SYSTEM

by

Jon D. Allen
and
Heidi R. Lang

September 1987

Thesis Advisor: Barry A. Frew

Approved for public release; distribution is unlimited

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION Unclassified	1b RESTRICTIVE MARKINGS
2a SECURITY CLASSIFICATION AUTHORITY 2b DECLASSIFICATION/DOWNGRADING SCHEDULE	3 DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; Distribution is Unlimited
4 PERFORMING ORGANIZATION REPORT NUMBER(S)	5 MONITORING ORGANIZATION REPORT NUMBER(S)

6a NAME OF PERFORMING ORGANIZATION Naval Postgraduate School	6b OFFICE SYMBOL (If applicable) Code 54	7a NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000		7b ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000
8a NAME OF FUNDING/SPONSORING ORGANIZATION	8b OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

8c ADDRESS (City, State, and ZIP Code)

10 SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO	PROJECT NO	TASK NO	WORK UNIT ACCESSION NO

11 TITLE (Include Security Classification)
SEARCH AND RESCUE AND ENFORCEMENT OF LAWS AND TREATIES INTELLIGENCE SYSTEM (u)

12 PERSONAL AUTHOR(S)
Allen, Jon D. and Lang, Heidi R.

13a TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Master's Thesis	FROM ______ TO ______	1987 September	430

16 SUPPLEMENTARY NOTATION

17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Database, Coast Guard, Law Enforcement & Search Rescue, SAR, Intelligence

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

In recent years, the-political climate surrounding drug interdiction has brought the focus of Coast Guard missions into emphasizing the prosecution of law enforcement cases. Many law enforcement cases are pursuant to a specific search and rescue case and traditionally, the Coast Guard has segregated its information reguarding vessels into separate systems for each of these areas. The existence of separate repositories for vessel data has resulted in duplication of information, lack of consistency, and the inability to integrate information between systems. The purpose of this thesis is to propose, design, impliment a prototype law enforcement/search and rescue database system to provide a single repository for law enforcement and search and rescue vessel data. The prototype will be developed at Coast Guard Group Monterey.

20 DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION Unclassified	
22a NAME OF RESPONSIBLE INDIVIDUAL Prof. Barry A. Frew	22b TELEPHONE (Include Area Code) (408) 646-2924	22c OFFICE SYMBOL Code 54Fw

83 APR edition may be used until exhausted
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

Approved for public release; distribution is unlimited.

Search and Rescue and Enforcement of Laws and Treaties Intelligence System

by

Jon D. Allen
Lieutenant, United States Coast Guard
B.S., United States Coast Guard Academy, 1979

and

Heidi R. Lang
Lieutenant, United States Navy
B.S., State University of New York at Buffalo, 1979

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS

from the

NAVAL POSTGRADUATE SCHOOL
September 1987

Authors: Jon D. Allen
Jon D. Allen

Heidi R Lang
Heidi R. Lang

Approved by: Barry A. Frew
Barry A. Frew, Thesis Advisor

Vincent Y L
Vincent Y. Lum, Second Reader

Willis R. Greer, Chairman,
Department of Administrative Science

James M. Fremgen,
Acting Dean of Information and Policy Sciences

ABSTRACT

In recent years, the political climate surrounding drug interdiction has brought the focus of Coast Guard missions into emphasizing the prosecution of law enforcement cases. Many law enforcement cases are pursuant to a specific search and rescue case and traditionally, the Coast Guard has segregated its information regarding vessels into separate systems for each of these areas. The existence of separate repositories for vessel data has resulted in duplication of information, lack of consistency, and the inability to integrate information between systems. The purpose of this thesis is to propose, design, and implement a prototype law enforcement/search and rescue database system to provide a single repository for law enforcement and search and rescue vessel data. The prototype will be developed at Coast Guard Group Monterey.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. INTRODUCTION

A. COAST GUARD GROUP MONTEREY

1. Overview

Coast Guard Group Monterey is the southernmost of three Groups in the Twelfth Coast Guard District. The Group is the primary Coast Guard interface with other federal, state, and local agencies and the general public in its area of responsibility. Group Monterey serves 175 miles of California coastline out to 60 nautical miles from Point Ano Nuevo (north of Santa Cruz) south to Point Sal (north of Santa Maria). Within this geographic area, the Group is tasked as the Search and Rescue Mission Coordinator.

The primary missions of Group Monterey include law enforcement, search and rescue, marine environmental protection, and boating safety. To fulfill these missions, the Group is staffed with over 100 men and women who operate and maintain a variety of law enforcement and rescue resources. The specific organizational structure and manning is outlined in Figure 1.1.

2. Search and Rescue (SAR)

Coast Guard Group Monterey has been an active SAR Mission Coordinator during the past eight years. From 1979 to 1983, there were over 1641 SAR cases. More detailed SAR statistics have been maintained since 1984 and are summarized for the period October 1, 1983 - April 1, 1987 in Ta [illegible]

Group Monterey has the following Coast Guard units available for SAR response:

- CG-41367 (41 foot utility boat)
- CG-41369 (41 foot utility boat)
- CG-44346 (44 foot motor lifeboat)
- CG-195003 (19 foot utility boat)
- Point Barrow (82 foot patrol boat)
- Point Heyer (82 foot patrol boat)
- Cape Wash (95 foot patrol boat)

Coast Guard and military aviation organizations and units supporting Group Monterey's SAR mission include:

- Coast Guard Air Station Sacramento

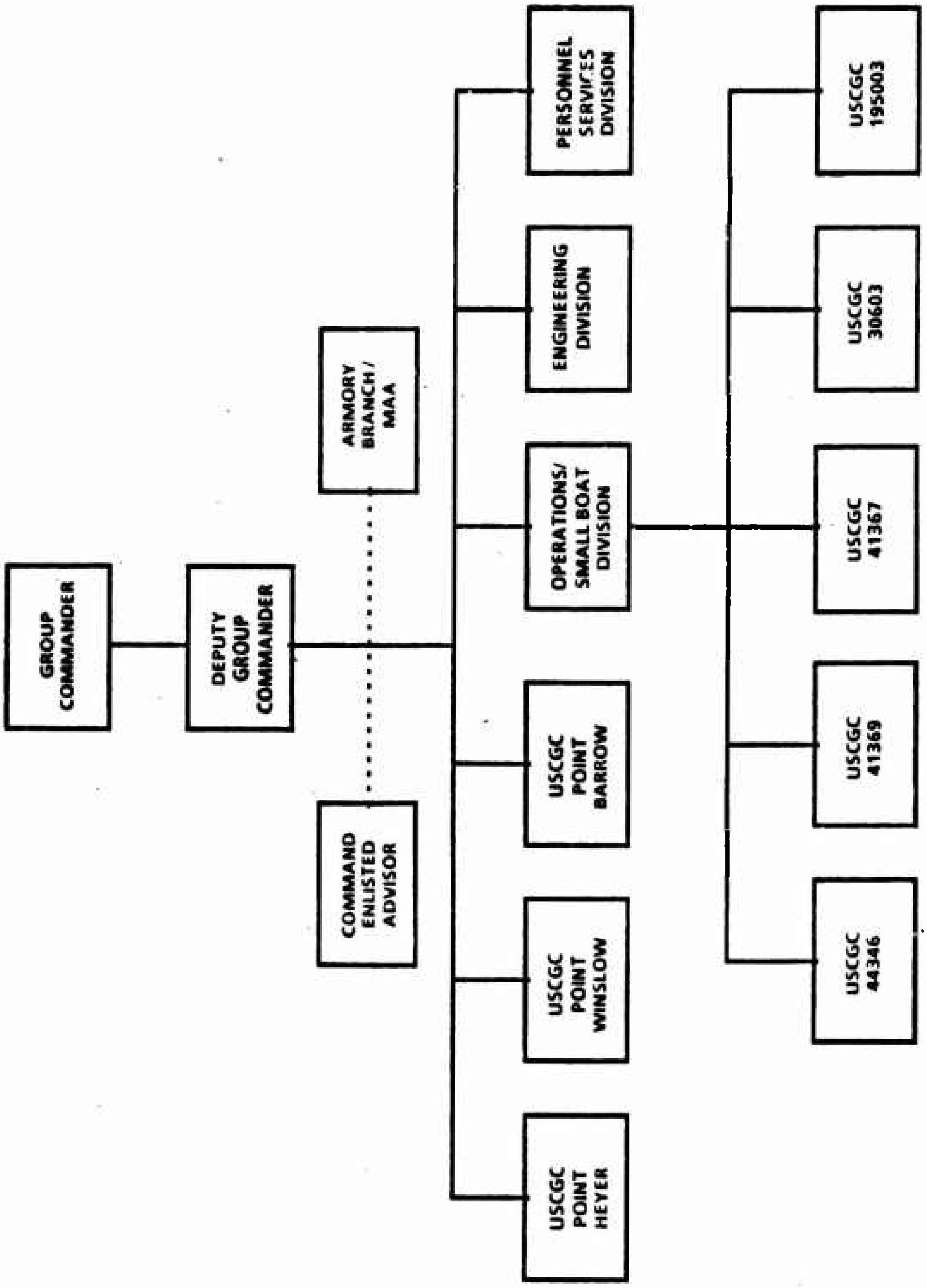


Figure 1.1 Group Monterey Organization.

- Coast Guard Air Station San Francisco
- Fort Ord MAST Helicopters
- Fort Hunter Ligget MAST Helicopters

Group Monterey, in addition to its Coast Guard and military SAR resources, has received SAR assistance from the following:

- Coast Guard Auxiliary vessels and aircraft
- Santa Cruz Harbor Master
- Moss Landing Boat Works
- California Fish and Game
- San Luis Obispo Harbor Master
- Morro Bay Harbor Master
- California Highway Patrol
- Various county sheriff departments
- Pacific Grove Marine Rescue
- Numerous local police units

TABLE 1
GROUP MONTEREY SAR STATISTICS

FY	Cases	Lives Lost	Lives Saved	Persons Assisted	Value of Property Lost	Value of Property Assisted
84	378	21	62	731	$ 2,033K	$ 111,473K
85	324	19	48	604	$ 1,403K	$ 7,492K
86	471	20	82	852	$ 616K	$ 10,217K
87	217	17	33	372	$ 122K	$ 5,276K

3. **Law Enforcement**

Group Monterey is mandated by law to conduct boardings of recreational, commercial, and passenger vessels to ensure compliance with applicable federal laws, regulations, and standards. The FY86 boarding statistics for Group Monterey are summarized in Table 2.

Agencies that assist or work in cooperation with Coast Guard Group Monterey include:

- Monterey County Sheriff's Department

- Monterey Police Department
- Monterey County District Attorney
- California Highway Patrol
- United States Customs Service (USCS)
- Bureau of Alcohol, Tobacco, and Firearms (ATF)
- Drug Enforcement Administration (DEA)
- California Department of Motor Vehicles
- Santa Cruz County Narcotics Enforcement Team
- United States Border Patrol
- Narcotic Detection Dogs (DELTA IX)
- Fort Ord Federal Police
- Santa Cruz Harbor Master
- Pacific Grove Police Department
- Morro Bay Police Department

TABLE 2

GROUP MONTEREY BOARDING STATISTICS

Total Boardings	444
Written Notices Issued.	181
Written Warnings Issued.	105
Safety Boardings Conducted.	158

B. STATUTORY AUTHORITY

The statutory authority that governs the operations of Coast Guard Group Monterey can be divided into two separate classifications. One area of responsibility and authority involves laws relating to navigation and merchant shipping. Laws that relate to customs and revenue, immigration, quarantine, protection of fish and game that fall within the jurisdiction of other federal agencies, but which require the unique authority and marine facilities of the Coast Guard for effective enforcement comprise the second legal category. Specific statutory authority which delineates and amplifies Coast Guard responsibilities and jurisdiction include the following:

1. 14 USC 2. This statute outlines the specific duties and responsibilities of the Coast Guard and gives the Coast Guard authority to enforce or assist in the enforcement of all applicable federal laws.
2. 14 USC 89. This statute grants specific law enforcement authority to the Coast Guard. It gives the Coast Guard the authority to make "inquiries, examinations, inspections, searches, seizures, and arrests on the high seas and waters over which the United States has jurisdiction for prevention, detection, and supression of violations of laws of the United States".
3. 14 USC 141. This statute gives the Coast Guard the authority to assist or be assisted by any federal agency, state, territory, or possession of the United States in the enforcement of applicable laws.
4. 14 USC 143 / 19 USC 1401(i). These statutes grant customs law enforcement authority to the Coast Guard.
5. 19 USC 1581. This statute grants authority to the Coast Guard, when enforcing customs laws, to seize any vessel, merchandise, or arrest any person in violation of these laws.
6. Port and Waterways Safety Act of 1970 (33 USC 1221). This act increases the supervision of vessel and port operations and mandates that all vessels operating in navigable waters of the United States comply with all applicable standards and requirements for vessel construction, equipment, manning, operational procedures, and safe handling of hazardous cargos.
7. Federal Boat Safety Act of 1971 (46 USC 1451). This act improves recreational safe boating and facilities by prescribing specific requirements and standards for boats and equipment.
8. Executive Order 11735 on Oil Pollution (August 3, 1973). This order directs the Coast Guard to receive notice of discharges of oil or hazardous substances, to establish procedures to prevent and contain discharges, to inspect vessels containing oil and hazardous substances and to board and arrest persons in violation of applicable laws.
9. Executive Order 11727 on Drug Enforcement (July 6, 1973). This order establishes a Drug Enforcement Administration within the Department of Justice and directs that each federal department and agency shall assist in the enforcement of laws involving narcotics and dangerous drugs.

C. THESIS BACKGROUND

As demonstrated throughout Coast Guard history and mandated by statute, the Coast Guard must be able to rapidly respond to any Search & Rescue crisis or Law Enforcement scenario. Additional and timely information in these crises and scenarios can make the difference between lives-saved and lives-lost, or violators apprehended and violators that escape prosecution. All too often, this information is maintained in filing cabinets --- virtually inaccessible when decisions must be made within minutes.

Coast Guard units, such as Group Monterey, must respond to situations where they have little or no information. To find out that a certain vessel owner/operator has a history of violations or is considered armed and dangerous can assist Coast Guard boarding officers in better preparing for what could be a life-threatening situation. Additionally, during SAR cases, it can be critical to know a vessel's description to facilitate its identification by land-based units, air resources, or Coast Guard vessels dispatched to search for the vessel in distress.

In order to respond to the need for automated information support, the Search and Rescue and Enforcement of Laws and Treaties Intelligence System (SARELTIS) was designed and implemented. Coast Guard Group Monterey served as the test bed for determining information needs and systems analysis. The system is able to not only respond to real-time SAR and Law Enforcement cases, but is also robust enough to allow users to analyze the data collected to form their own local intelligence information base. The system's standard reports can provide limited decision-making support for Coast Guard resource allocation and scheduling.

D. GOALS AND OBJECTIVES

The goals and objectives of SARELTIS are twofold: first, to provide a real-time query capability and second, to provide historical management reports that can aid in resource allocation, scheduling, and distribution. The following functions/products will be performed/ prepared by SARELTIS:

a. Interactive Query
   * Vessel Searches
   * Personnel Searches

b. Intelligence Reporting
   * El Paso Intelligence Center (EPIC) Intelligence Summary
   * Vessel Profiles for Violation Analysis
   * Personnel Intelligence Summary

c. Boarding Historical Reports
   * Violation/Unsafe Condition Analysis
   * Courtesy Motorboat Examination (CME) Analysis
   * Geographic Area Analysis

d. SAR Historical Reports
   * Immediate SAR Case Analysis
   * Geographic Area Analysis

- SAR Unit Operational Time Utilization Summary
- Days/Months/Years SAR Case Analysis

e. Vessel Historical Reports

- Vessel Summary

f. Personnel Historical Reports

- Vessel Owners Summary
- Vessel Operators Summary

E. METHODOLOGY

Two prototypes were developed as a precursor to SARELTIS. Functionally, the first prototype concentrated on the Law Enforcement information requirements on a limited scale while the second prototype concentrated on SAR. After proving the technical feasibility of the SARELTIS concept, a feasibility study was prepared and is included as Chapter II. The two prototypes will be discussed in Chapter III.

The logical design structure necessary to fulfill the information requirements to fully implement SARELTIS is presented in the data flow diagrams of Appendix A. These dataflow diagrams depict the complete analysis of the SAR and Law Enforcement functions and processes of Coast Guard Group Monterey, and were developed through extensive interviews with Group personnel. Additionally, these diagrams show the automation boundary of SARELTIS, which is a graphic representation of precisely where SARELTIS will fit into these mission areas. Once this automation boundary was identified, the logical entities and their relationships within this boundary were determined and represented by the Bachman diagram found in Appendix A.

The physical design process for the SARELTIS prototype involved the conversion of the logical model into a language-specific database structure and an identification of the modules required to implement the required functions of the application. To perform this conversion, all of the currently used source documents were identified. A survey was prepared containing all of the data elements present on these forms and was distributed to representative groups of users at Group Monterey. These users identified the degree to which the data elements were necessary for the performance of their duties. From the results of this survey, the data elements required within the SARELTIS database were identified. These data elements were then grouped within the previously defined entities of the Bachman diagram and became the basis for the SARELTIS physical database structure.

At the completion of physical database structure implementation, the specific functions that were required to operate upon this structure were identified, including the generation of all required and anticipated user and management reports. These functions were then divided into implementable modules, coded, and tested. Upon completion of testing, modules with related functionality were combined and the submenu code required to invoke each application was developed. Finally, the main menu, security, and utility modules were constructed and the entire system was integrated. The module functional descriptions are provided in Appendix D and the menu hierarchy charts are included in Appendix C.

The User's Guide, located in Chapter IV, provides the user with a step-by-step guide to the SARELTIS menu-based system. The various reports generated by SARELTIS are displayed in Appendix B.

F. SYSTEM REQUIREMENTS

To operate SARELTIS, the following hardware is required:

* IBM® PC-XT, PC-AT, or compatible microcomputer
* DOS®, Version 2.0 or higher
* A hard disk and one double-sided, double-density 5.25 floppy disk drive
* 384K of main memory (system processing is greatly enhanced if 640K or 1M of main memory is available)
* Color or monochrome monitor
* Printer, with normal or large carriage

SARELTIS was developed utilizing the R:base 5000® Database Management System written by MicroRim, Inc., of Belleview, Washington.

G. SYSTEM OVERVIEW

SARELTIS is an interactive, user-friendly, menu-driven system. A series of screens displaying various menus provide the user with a functional man-machine interface.

The overall system design is based upon one relational database accessed by eleven functionally separate application modules. Application modules are comprised of functionally distinct programs. These program functions are listed in Chapter IV, the User's Guide, with each module's function explained in detail in Appendix D.

All system reports are designed to be printed on a normal (small) width carriage printer. The forty five reports prepared by the system were designed to assist the

Commanding Officer or Operations Officer to make better resource and planning decisions.

The R:base 5000 language is easily accessible for the advanced user and can be used to build other applications, reports, forms, searches, and displays of SARELTIS information not covered in the original implementation.

II. SARELTIS FEASIBILITY

A feasibility study is a compressed, capsule version of the entire systems analysis and design process. As part of the feasibility process, the existing system is examined, a high level model of the proposed system is developed, and the overall feasibility of the project is evaluated. This chapter will examine the economic and the political feasibilities of SARELTIS. The technical feasibility will be addressed in Chapter III (discussion of the prototypes).

A. POLITICAL FEASIBILITY

Due to recent Congressional pressures and unfavorable publicity concerning the Coast Guard's zealous boarding practices in unwarranted situations, attention has been focused on the lack of available information provided to boarding officers prior to the boarding of a vessel. Boarding officers must comply with the provisions of Title 14 of the U.S. Code, Section 89, concerning their use of force necessary to compel public and commercial boaters to comply with U.S. laws and boarding directions. However, without adequate information or intelligence prior to boarding, the degree of initial force necessary to ensure this compliance is difficult if not impossible to predetermine. Therefore, any means of providing this intelligence information would be strongly welcomed at all levels of the Coast Guard.

Coast Guard Group Monterey was contacted in early June 1986 to determine their willingness to serve as a test site for the implementation of a prototype intelligence system. Their response was overwhelming and highly supportive. Following the initial positive reaction of Group Monterey, both the Twelfth Coast Guard District and Coast Guard Pacific Area intelligence offices were approached in December 1986 concerning the proposal and the results of the first prototype. They also displayed a high degree of enthusiasm for the effort. In April 1987, RADM W. Merlin, the Coast Guard's senior management responsible for all computer applications, expressed an interest in the project and promised additional hardware support.

It is obvious from the above that the concept of providing automated intelligence and information is supported at all levels of the Coast Guard --- from local operational personnel to top management of the entire organization. This kind of multi-level organizational support has shown that the SARELTIS project is politically feasible.

B. ECONOMIC FEASIBILITY

Economic feasibility is demonstrated if it can be shown that the benefits of a system outweigh its costs. Before making any cost/benefit study, it is important to list the assumptions that were used so that the reader may see what constraints were imposed on the analysis and where the costs and benefits were derived from. The following is a list of assumptions used in formulating the cost/benefit analysis:

a. Management information research on Search and Rescue (SAR) cases provided monthly.

b. Management information research on Enforcement of Laws and Treaties (ELT) cases provided weekly.

c. SAR management information research performed by Group Monterey's Officer-of-the-Day (paygrade E-6).

d. L/E management information research performed by Group Monterey's ELT Specialist (paygrade E-5).

e. Specific ELT case research performed by Group Monterey's Officer-of-the-Day (OOD).

f. Specific SAR case research performed by Group Monterey's Officer-of-the-Day (OOD).

g. Assume 2,000 man-hours per year.

h. Development of SARELTIS performed by two lieutenants (paygrade O-3).

i. All costs and benefits were evaluated based on the Coast Guard's Annual Standard Personnel Costs [Ref. 1]. The pertinent costing data is as follows:

   1) paygrade E-5 --- $18,500 ($9.25 / hour)

   2) paygrade E-6 --- $22,300 ($11.15 / hour)

   3) paygrade O-3 --- $33,900 ($16.95 / hour)

j. The annual SAR case load is 386 (Table 1).

k. The annual ELT case load is 444 (Table 2).

l. The annual SAR and ELT case loads will remain relatively constant over the expected life of the project.

m. SARELTIS has an expected 5 year project life.

n. Approximately three years of historical SAR and ELT case data will be entered upon system implementation.

In order to better understand how the actual computations contained in this section were derived, a synopsis of both the existing and proposed systems will be discussed below.

1. Existing Method

ELT and SAR case load research is currently not conducted due to the amount of time necessary to perform the research and the inaccessibility of the information. It has been estimated by Coast Guard Group Monterey OOD's that such research would take a minimum of 1.5 to 2.0 hours to search through their existing records to locate any available pertinent information if the information could be found at all.

Management information research on ELT cases is performed by the Group's ELT Specialist (paygrade E-5) who must manually compile statistics on a weekly basis. He estimates that it takes a minimum of 1 man-day to perform the required research and the preparation of subsequent required reports.

Management information research on SAR cases is performed by the Group's OOD (paygrade E-6) who must manually compile statistics on a monthly basis. It is estimated that this research takes a minimum of 3 man-days to perform the research and to prepare the required reports.

2. Proposed System

ELT and SAR specific case research will be handled through customized database query program modules. It is estimated each query would take an average of 3 minutes to enter the search criteria and perform the actual search.

Management information research on ELT cases would be done through the use of customized reports where users can define specific timeframes, vessels, violations, or personnel to analyze or search. It is estimated that it would take approximately 30 minutes to perform a series of reports relating to ELT information.

Management information research on SAR cases would be done through the use of customized reports where users can define specific timeframes, vessels, or personnel to analyze or search. It is estimated that it would take approximately 30 minutes to perform a series of reports relating to SAR information.

3. Existing Method Analysis

Recurring Costs:

SAR Case Load Research:

Filling out required SAR case documentation:	
2 hours x $11.15 (paygrade E-6)	$22.30
Annual SAR case load	386
	$8,607.80
Retrieving specific SAR case data:	
1.5 hours x $11.15 (paygrade E-6)	$16.73
Annual SAR case load	386
	$6,457.78
Total SAR Case Load Research:	$15,065.58

ELT Case Load Research:

Filling out required ELT case documentation:	
1 hour x $11.15 (paygrade E-6)	$11.15
Annual ELT case load	444
	$4,950.60
Retrieving specific ELT case data:	
1.5 hours x $11.15 (paygrade E-6)	$16.73
Annual ELT case load	444
	$7,428.12
Total ELT Case Load Research:	$12,378.72

SAR Management Information Research:

Researching SAR documentation:	
4 man-days x $11.15/hour x 8 hours (paygrade E-6)	$356.80
Monthly Research	12
Total SAR Management Information Research	$4,281.60

ELT Managment Information Research:

Researching ELT documentation:	
1 man-day x $9.25 x 8 hours (paygrade E-5)	$74.00
Weekly Research	52
Total ELT Management Information Research	$3,848.00

Total Existing Method Costs (Recurring):

Total SAR Case Load Research:	$15,065.58
Total ELT Case Load Research:	$12,378.72
Total SAR Management Information Research	$4,281.60
Total ELT Management Information Research	$3,848.00
Total Recurring Costs:	$35,573.90

4. **Proposed Method Analysis:**

Non-Recurring Costs:	
First Prototype Development (using R:base 4000):	
250 man-hours x $16.95 (paygrade O-3)	$4,237.50
Second Prototype Development (using ReQuest):	
200 man-hours x $16.95 (paygrade O-3)	$3,390.00
SARELTIS Development (using R:base 5000):	
2000 man-hours x $16.95 (paygrade O-3)	$33,900.00
SARELTIS Initial Historical SAR Data Loading:	
.25 hours x $11.15 (paygrade E-6)	$2.79
386 cases x 3 years	1,158
	$3,230.82
SARELTIS Initial Historical ELT Data Loading:	
.25 hours x $11.15 (paygrade E-6)	$2.79
444 cases x 3 years	1332
	$3,716.28
Hardware / Software Costs:	
IBM PC-XT compatible computer and printer	$2,000.00
R:base 5000 software	$400.00
	$2,400.00
Total Non-Recurring Costs:	$50,874.60

Recurring Costs:

SAR Case Load Research:

Filling out required SAR case documentation:	
2 hours x $11.15 (paygrade E-6)	$22.30
Annual SAR case load	386
	$8,607.80
Loading SARELTIS with SAR case data:	
.25 hours x $11.15 (paygrade E-6)	$2.79
Annual SAR case load	386
	$1,076.94
Retrieving specific SAR case data:	
.05 hours x $11.15 (paygrade E-6)	$0.56
Annual SAR case load	386
	$216.16
Total SAR Case Load Research:	$9,900.90

ELT Case Load Research:

Filling out required ELT case documentation:	
1 hour x $11.15 (paygrade E-6)	$11.15
Annual ELT case load	444
	$4,950.60
Loading SARELTIS with ELT case data:	
.25 hours x $11.15 (paygrade E-6)	$2.79
Annual ELT case load	444
	$1,238.76
Retrieving specific ELT case data:	
.05 hours x $11.15 (paygrade E-6)	$0.56
Annual ELT case load	444
	$248.64
Total ELT Case Load Research:	$6,438.00

SAR Management Information Research:	
Researching SAR documentation:	
.5 hours x $11.15 (paygrade E-6)	$5.58
Weekly Research	12
Total SAR Management Information Research	$66.96
ELT Managment Information Research:	
Researching ELT documentation:	
.5 hours x $11.15 (paygrade E-6)	$5.58
Weekly Research	12
Total ELT Management Information Research	$66.96
System Manager Data Validation:	
100 hours x $11.15 (paygrade E-6)	$1,115.00
Hardware Corrective Maintenance:	$500.00
Software Corrective Maintenance:	$500.00
Total Recurring Costs:	$18,587.82

5. **Proposed versus Existing Method Cost Comparisons:**

Total Existing Non-Recurring Costs	$0.00
Total Proposed Non-Recurring Costs	$50,874.60
Additional Non-Recurring Costs	$50,874.60

Total Existing Recurring Costs	$35,573.90
Total Proposed Recurring Costs	$18,587.82
Total Recurring Savings	$16,986.08

Payback Period [Ref. 2: p. 319]:

First Year Costs	$50,874.60
First Year Benefits	$16,986.08

Payback Period = Costs / Benefits
= $50,874.60 / $16,986.08
= 2.99 years

Internal Rate of Return (IRR):

Estimated First Year Costs		$50,874.60
Estimated First Year Benefits	$16,986.08	
Estimated Second Year Benefits	$16,986.08	
Estimated Third Year Benefits	$16,986.08	
Estimated Fourth Year Benefits	$16,986.08	
Estimated Fifth Year Benefits	$16,986.08	
Calculated IRR [Ref. 2: p. 320-322]:		19.93%

III. TECHNICAL FEASIBILITY

Can the system be implemented using current technology [Ref. 2: p. 274]? This is the question that must be answered to determine and evaluate the technical feasibility of a project. For the SARELTIS project, the issue of current technology was approached from two different perspectives. First, was the concept of a historical database system that could provide on-line query information and complex management reports an implementable concept within the scope of a thesis project? And second, was this concept implementable within the constraints of the Coast Guard standard database language, ReQuest®? During the last year, two limited scale prototypes were developed by the authors to answer these two questions of project technical feasibility.

A. LAW ENFORCEMENT INFORMATION SYSTEM

Concept implementability was the question that was addressed by the design of the Coast Guard Law Enforcement Information System, the first prototype. This prototype was programmed on the Coast Guard Standard Terminal System using the relational database system R:base 4000®, and provided historical information on vessels, operators, and vessel boardings. It was designed to facilitate the retrieval of "intelligence-type" information to better prepare boarding officers and boarding petty officers prior to stepping onto a vessel. It also provided statistical information of the locations, times, dates, and the types of vessels that were boarded.

The logical design of this prototype included the entities of an operator, a vessel, and a boarding. To implement this logical design, additional relations were added as required to provide the necessary intersection mechanisms, rules, forms, reports and editing constraints. All restrictions concerning the legal values of attributes and interrelational constraints are stated and controlled through the use of the R:base 4000 rules feature. Menu chaining was used to provide a user friendly system, and menus and programs were written in and controlled through the use of the R:base 4000 command language. The R:base 4000 environment proved to be very capable of supporting this limited scale application.

This prototype was successfully demonstrated in December 1986. The prototype demonstrated the capability of complex nautical chart searching algorithms for use in making key management decisions for "high incident areas" and "high incident times" and the capability of searching the database for vessels of interest within a given search criteria.

Therefore, through the use of the Law Enforcement Information System prototype, the question of concept implementability was successfully satisfied.

B. SEARCH AND RESCUE INTELLIGENCE INFORMATION SYSTEM

Since all Coast Guard "production" database management system implementations are required to use the Coast Guard standard database language, ReQuest, another limited scale prototype was developed to assess the feasibility of application development within the constraints imposed by Coast Guard directives. The development and testing of this prototype, the Search and Rescue (SAR) Intelligence Information System, occurred during June 1987.

The Search and Rescue Intelligence Information System was programmed on the Coast Guard Standard Terminal System using the language ReQuest, and provided historical information on vessels, owners, and vessel search and rescue cases. It was designed to facilitate the retrieval of detailed vessel and owner historical information to better enable the SAR case personnel to conduct the SAR mission.

The entities of an owner, a vessel, and a SAR case were represented within the logical structure of this prototype. The physical structure included an intersection record to link the owner and vessel entities. All attribute and interrelational constraints were specified and controlled through the attribute descriptions. Forms and reports were created and stored as individual files, independent of the database structure. Menus were very limited as ReQuest does not allow menu chaining beyond the top level. Reports were also very limited due to the restrictions imposed by ReQuest concerning multiple relation searching and reporting. The system was designed to replicate the functionality of the first prototype as much as possible, however, the ReQuest environment did not allow this parallelism.

C. INADEQUACIES OF REQUEST FOR THE SARELTIS PROTOTYPE

Numerous problems were encountered during the development of the second prototype that were directly attributable to the use of the language ReQuest. This language is very limited in scope and inflexible in nature. It does not support any type of command language for program or menu construction. To successfully utilize an application developed within this environment, the user must be very knowledgeable and fully understand both the language and the application. Unlike more flexible programming environments, the details of the language and the underlying structure of the application cannot be made transparent to the user. Additionally, the true strengths of a relational system, namely the ability to join and search multiple tables for reports or ad-hoc queries, is absent in this language. Reports are limited to the use of two tables at most, with certain restrictions, and information requested by ad-hoc queries may only come from a single predefined table. Any tables that result from a relational operation, such as a join, cannot then be searched by an adhoc query. Therefore, to provide the desired searching functionality within the SARELTIS prototype, the physical implementation would have to consist of a single relation, and would be subject to all the anomalies inherent in a first normal form relational database [Ref. 3: p. 286-290]. Speed is also a major problem. To accomplish even the simplest tasks required a great deal of waiting time on the part of the user. Finally, the security system provided by the language did not function properly. As a result of these constraints imposed by the ReQuest language, it was determined that a different environment would have to be used to successfully implement the SARELTIS prototype.

D. PROPOSED LANGUAGE ENVIRONMENT

Since ReQuest was determined to be infeasible as a language environment for the SARELTIS project, the additional languages available for the Coast Guard Standard Terminal System were examined to determine if any were suitable. At present, the only additional language available for the standard terminal and the CTOS operating system is the R:base 5000 relational database environment. As this is an updated and enhanced version of the language used for the first prototype, it was determined that R:base 5000 would be used for the SARELTIS project. Unfortunatly, the R:base 5000 environment was not available locally for the Coast Guard Standard Terminal System. The language was, however, available for the IBM DOS operating system. Therefore, the decision was made to develop the SARELTIS prototype utilizing the R:base 5000 environment within the IBM DOS operating system. This decision allowed the development of a fully capable product that could ultimately be converted for use on the Coast Guard Standard Terminal System, and established full technical feasibility for the SARELTIS prototype.

IV. USERS GUIDE

A. INTRODUCTION

The following user's guide is written assuming that its intended user will be utilizing an IBM PC-XT or an IBM compatible computer with a 5.25 inch floppy disk drive, a minimum of 640K internal memory, a hard disk (minimum 10M Bytes), a printer with a normal size carriage, and using the DOS operating system. If a different hardware configuration is used, the user should refer to the owner's manual to see how to load R:base 5000 and to determine if the configuration will support the database system and SARELTIS.

SARELTIS was designed to assist Coast Guard Group Monterey in automating historical information and intelligence retrieval. To this end, SARELTIS provides comprehensive input and modify programs to assist the user in easily entering information and correcting/updating existing information in the system. Additionally, a realtime query module allows users to access system database records quickly and easily in order to provide the most current intelligence and historical information in a timeframe that can be used in operational missions.

This user's guide is an attempt to direct users through the major portions of SARELTIS. A recent thesis by LCDR J. M. Graham [Ref. 4] provided some of the framework and content of this section. Due to the actual size of the SARELTIS program and the desire to keep the user's guide short and easy to understand, the technical inner workings of each program will not be discussed. Users desiring to know more information should consult the R:base 5000 technical manuals or contact one of the authors for additional clarification. The following topics will be addressed to assist the user with SARELTIS operations:

- System Startup Procedures
- Security
- Operations
- Exiting
- Maintenance

B. SYSTEM STARTUP PROCEDURES

1. Machine Configuration Requirements

Due to the size of SARELTIS (over 800K in database structure and over 500K in application program modules), the user must ensure that the machine is properly initialized with a CONFIG.SYS file which contains the following:

- Buffers = 5
- Files = 20

If this file does not exist or does not contain the information listed above, the user will need to build or modify the CONFIG.SYS file accordingly. The user should consult the DOS Manual for the specific procedure necessary to create or alter a CONFIG.SYS file. Once the CONFIG.SYS file has been properly created/modified, the computer should be reset/booted to allow this new machine configuration to be installed. To reset the system, the user presses the following three keys simultaneously: <CTRL> <ALT> <DEL>. After the machine has been reset, the computer will ask the user for the current date and current time. It is very important that the user enter the correct time and date into the system as the system time and date are used by many application program modules. After each date and time entry, the user should press the <ENTER> key.

2. Initial Program Loading

If all has proceeded satisfactorily from the above section, the screen should display the following system prompt indicating that you are working from the hard disk: "C:>".

Use the DOS MKDIR (make directory) command to create a directory to store the R:base 5000 programs by typing "MKDIR C:\R5K" at the system prompt. Next, use the DOS CHDIR (change directory) command to change the working directory to the newly created R:base 5000 directory by typing "CHDIR C:\R5K" at the system prompt. At this point you are ready to load the R:base 5000 software if it is not already been installed. To load the R:base 5000 software [Ref. 5: p. 1.3 - 1.6], do the following:

1. Insert one of the R:base 5000 disks into floppy drive a.
2. Copy all files from the disk to the newly created R:base 5000 directory by entering "COPY *.*".
3. When all of the files have been copied from the floppy disk to the R:base 5000 directory on your hard disk, remove the floppy disk from the floppy drive.

4. Repeat steps 2 through 4 until all R:base 5000 floppy disks have been copied to your hard disk.

After you have loaded the R:base 5000 database software, you are now ready to load the SARELTIS application. SARELTIS has been designed to operate from its own directory (DBDIR) which the user must create. To create another directory, use the DOS MKDIR command and type "MKDIR C:\DBDIR". Next, you must copy the SARELTIS database and its application and ancillary programs to the DBDIR directory.

The SARELTIS database is stored on 4 diskettes labeled:

- SARELTIS 1
- SARELTIS 2
- SARELTIS 3
- SARELTIS 4

The application programs are stored on 2 diskettes labeled:

- SARELTIS APPLICATIONS 1
- SARELTIS APPLICATIONS 2

The source code for the applications programs are stored on 2 diskettes labeled:

- SARELTIS APPLICATIONS SOURCE CODE 1
- SARELTIS APPLICATIONS SOURCE CODE 2

The application program source code disks are provided for maintenance purposes only. The ancillary programs are stored on 1 diskette labeled SARELTIS ANCILLARY PROGRAMS. To load SARELTIS, do the following:

1. Insert SARELTIS 1 into the A: floppy drive.
2. Type the following DOS command: "RESTORE a: \DBDIR ". You will be prompted by the system to load SARELTIS 2, 3, and 4.
3. Insert SARELTIS APPLICATIONS 1 into the A: floppy drive.
4. Type the following DOS command: "RESTORE a: \DBDIR ". You will be prompted by the system to load SARELTIS APPLICATIONS 2.
5. Insert the SARELTIS ANCILLARY PROGRAMS floppy disk into the A: floppy drive.
6. Type the following DOS command: "COPY a:*.*".

You have now loaded the SARELTIS database and its application programs and are now ready to start the program. To start SARELTIS, enter the following at the system prompt: "SARSYS". This will automatically take you to a security module that will ask for a user name and password. The default user name is "SYSOP" and the default user password is "SARSYS". If you wish to protect the system, see the next section on security.

C. SARELTIS SECURITY

One of SARELTIS' ancillary programs is called "STARTUP.BIN". This program is a compiled/binary program which prints the initial SARELTIS title screen and provides the security module to gain access to the system. As discussed in the previous section, the system has a default user name and password. These can be changed very easily by the user (it is recommended that the user become somewhat familiar with R:base 5000 and SARELTIS before changing the security). Using a text editor or a word processor in the non-document mode, open the ancillary program "STARTUP.PRG". Approximately halfway through the program, you will find the following command:

"if username eq "SYSOP" and userpass eq "SARSYS" then"

By changing "SYSOP" to another user name and altering "SARSYS" to another password, you can protect your own system. Once you have changed the user name and password, ensure that you write it down and store it in a safe place. To activate the new security password, you must recompile "STARTUP.PRG". To recompile STARTUP.PRG, do the following:

1. At the system prompt, type "RCOMPILE" to run the R:base 5000 program compiler. The system will then display the RCOMPILE menu listing eight possible options.
2. Select the option to "Convert an ASCII command file to a binary command file" by entering a "1". The cursor should highlight the "(1)" next to the desired menu option.
3. Press the <ENTER> key to execute the menu option.
4. The system will ask you to enter the name of the ASCII command file to convert. Enter "STARTUP.PRG" and press the <ENTER> key.
5. RCOMPILE will then respond asking for the name of the backup file. Since we do not need a backup of this file, press the <ENTER> key.
6. The system will then ask for the name of the binary command file that you wish to create. In the space provided, enter "STARTUP.BIN" and press the <ENTER> key. The system will now re-compile the program. At the completion of the compilation, RCOMPILE will return you to its menu screen. Enter an "8" and press <ENTER> to exit from the RCOMPILE program.

You have now changed the security on the system and the new user name and password have been activated. To ensure that the system is fully secure, remove the

"STARTUP.PRG" file by typing the following DOS command: "ERASE STARTUP.PRG". This will prevent any unauthorized users from discovering the new security password by merely typing out the "STARTUP.PRG" file. If you wish to alter the security password after this point, you must copy the "STARTUP.PRG" file to the hard disk from the SARELTIS APPLICATIONS SOURCE floppy disk.

D. SARELTIS OPERATIONS

1. Menu System Overview

The man-machine interface for SARELTIS is a series of menus. Beginning with the SARELTIS Master Menu (Figure 4.1), the user can branch to a desired function. Each menu allows the user to return to the previous menu. All menu selections are grouped by functional area to provide the user with a more efficient operating environment. Examples of the remaining menu screens are presented in Figures 4.2 through 4.4 and Figures 4.7 through 4.13 Detailed functional descriptions of each menu option can be reviewed by examining Appendix D.

2. Starting SARELTIS

To start SARELTIS and to get to the main menu screen requires the user to type "SARSYS" at the system prompt and pressing the <ENTER> key. As described in the security section above, the user must supply the correct user name and password in order to gain access to the system. Once the proper user name and password are entered, the system is automatically initialized and the SARELTIS Master Menu appears. The user is free to select any option by either moving the cursor up/down (using the arrow keys on the numeric pad) or by typing the actual number of the option desired to be executed. Once a selection is made, pressing the <ENTER> key will execute the option. The user will then be presented with another menu screen or an actual function will be performed. Short descriptions of each menu option are presented in the following section.

3. SARELTIS Menu Options

a. SARELTIS Master Menu (Figure 4.1)

Selecting option (1), INPUT INFORMATION, displays the SARELTIS INPUT MENU (Figure 4.2). Option (2) calls the SARELTIS MODIFY MENU (Figure 4.3). The real-time query menu, QUERY INFORMATION MENU (Figure 4.4), is displayed upon selection of choice (3). The REPORTS MAIN MENU (Figure 4.7) is called when option (4) is chosen. Option (5) provides the user with the UTILITIES MENU (Figure 4.13). User-designed queries or reports may be constructed by selecting option (6), AD HOC QUERIES, which returns the user to the R:base 5000 command prompt "R>". To exit SARELTIS, the user selects option (7), EXIT SARELTIS.

```
                    SARELTIS MASTER MENU
(1)     INPUT INFORMATION
(2)     MODIFY INFORMATION
(3)     QUERY INFORMATION
(4)     REPORTS
(5)     UTILITIES
(6)     AD HOC QUERIES
(7)     EXIT SARELTIS
```

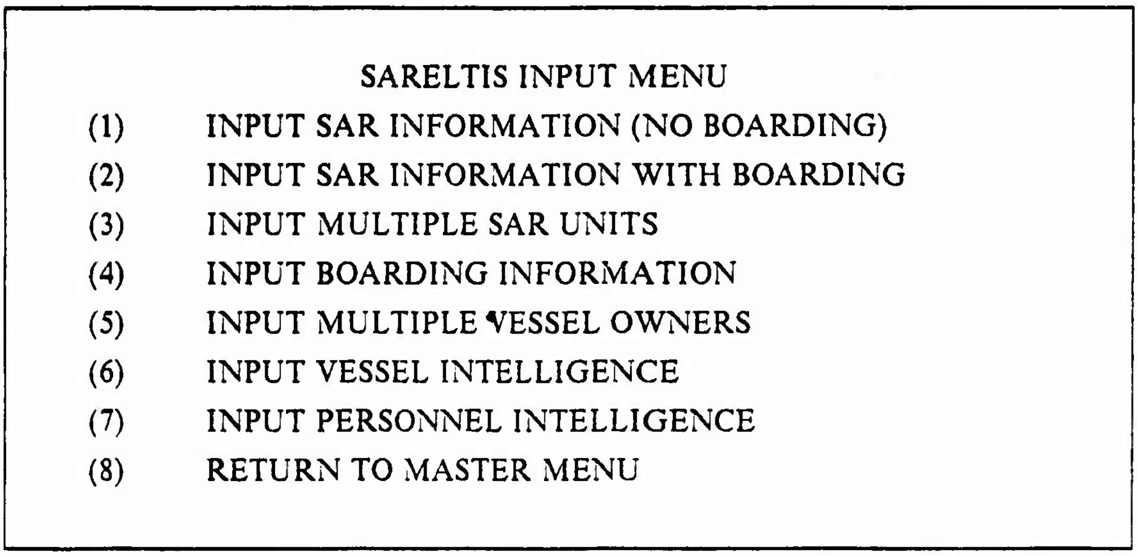
Figure 4.1 SARELTIS Master Menu.

b. Input Information Menu (Figure 4.2)

The user is provided with eight options to enter data into SARELTIS. The user has the choice of entering SAR information alone by selecting option(1) or entering SAR information with boarding information by choosing option (2). Option (3) allows the user to enter in multiple SAR units. Entering in boarding information alone is performed by choosing option (4). Selecting option (5) will allow the user to enter in additional vessel owners. Vessel intelligence and personnel intelligence are entered into the system by selecting options (6) or (7). To return to the SARELTIS MASTER MENU, the user selects option (8).

c. SARELTIS Modify Menu (Figure 4.3)

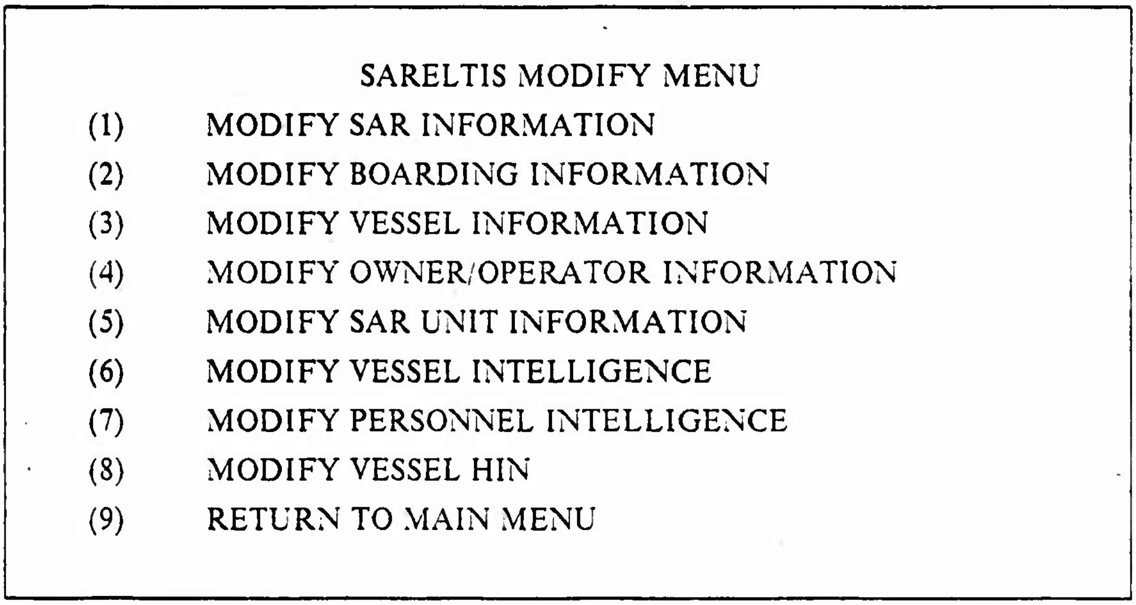
SARELTIS data can be modified using one of the nine available options. Option (1) allows the user to update historical SAR information. By selecting option (2), existing boarding information can be modified. Vessel information may be changed by selected option (3). Demographic data maintained on vessel owners and operators can be altered by choosing option (4). If the user desires to change the name of an owner or an operator and this individual is both an owner and an operator, the name must be changed in both places. Information regarding Group Monterey SAR



```
                    SARELTIS INPUT MENU
(1)      INPUT SAR INFORMATION (NO BOARDING)
(2)      INPUT SAR INFORMATION WITH BOARDING
(3)      INPUT MULTIPLE SAR UNITS
(4)      INPUT BOARDING INFORMATION
(5)      INPUT MULTIPLE VESSEL OWNERS
(6)      INPUT VESSEL INTELLIGENCE
(7)      INPUT PERSONNEL INTELLIGENCE
(8)      RETURN TO MASTER MENU
```

Figure 4.2 SARELTIS Input Menu.

units can be changed by selecting option (5). Options (6) and (7) allow the user to manipulate vessel and personnel intelligence information. The vessel HIN may be changed by selecting option (8) if it was previously entered incorrectly. To return to the MASTER MENU, the user selects option (9).

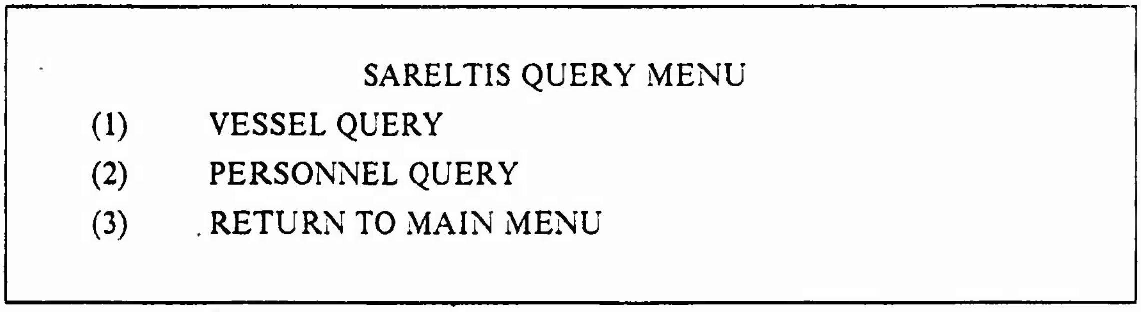


```
                    SARELTIS MODIFY MENU
(1)      MODIFY SAR INFORMATION
(2)      MODIFY BOARDING INFORMATION
(3)      MODIFY VESSEL INFORMATION
(4)      MODIFY OWNER/OPERATOR INFORMATION
(5)      MODIFY SAR UNIT INFORMATION
(6)      MODIFY VESSEL INTELLIGENCE
(7)      MODIFY PERSONNEL INTELLIGENCE
(8)      MODIFY VESSEL HIN
(9)      RETURN TO MAIN MENU
```

Figure 4.3 SARELTIS Modify Menu.

d. SARELTIS Query Menu (Figure 4.4)

The real-time query capability of SARELTIS is executed through the invocation of this menu. By selecting Option (1), the user can perform a vessel query based on vessel Hull Identification Number (HIN), vessel name, vessel number, or vessel official number. The query module will search the entire vessel table attempting to find potential matches to the user's selection criteria. Figure 4.5 shows the chaining of searches through the database when selecting the vessel query option. Similarly, the user can search through the database using a person's last name and first name by choosing option (2). The query program will find all potential matches and ask the user to select the correct individual to perform an intelligence/history check. Figure 4.6 shows the chaining of searches through the database for the personnel query option. To return to the SARELTIS MASTER MENU, the user selects option (3).



SARELTIS QUERY MENU

(1) VESSEL QUERY
(2) PERSONNEL QUERY
(3) RETURN TO MAIN MENU

Figure 4.4 SARELTIS Query Menu.

e. SARELTIS Reports Main Menu (Figure 4.7)

Reports in SARELTIS are broken down into mission/functional area groups that can be selected from the Reports Main Menu. This menu calls other sub-menus which represent the available choices within the specific mission/functional area. Selecting option (1) will allow the user the display the INTELLIGENCE REPORTS MENU (Figure 4.8). Option (2) will invoke the BOARDING REPORTS MENU (Figure 4.9). The SAR REPORTS MENU (Figure 4.10) is executed by choosing option (3). Option (4) will call the VESSEL REPORTS MENU (Figure 4.11). By selecting option (5), the user will see the available choices from the PERSONNEL REPORTS MENU (Figure 4.12). Detailed descriptions of each report function can be found in Appendix D. To return to the MASTER MENU, the user selects option (6).

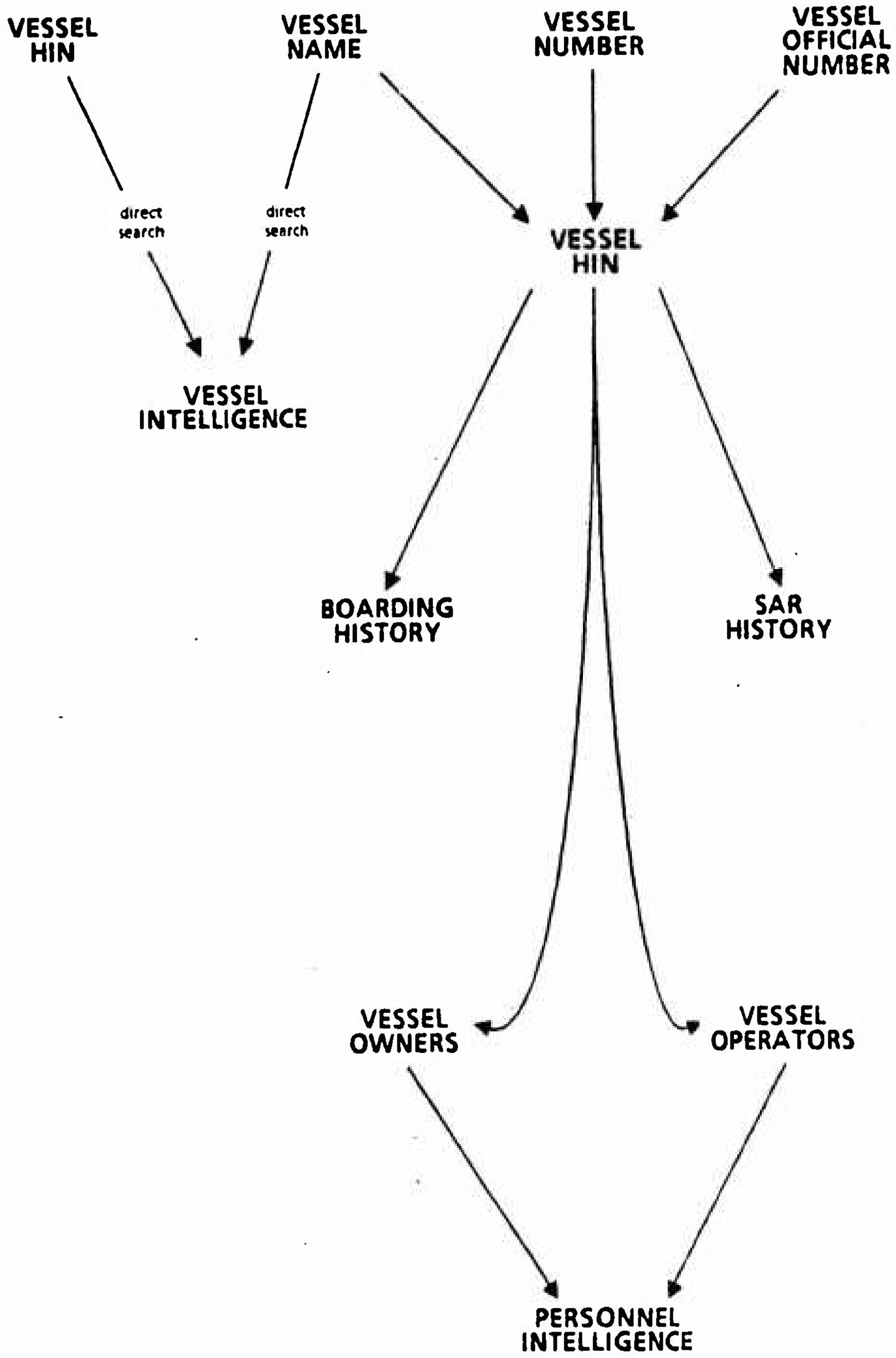


Figure 4.5 Vessel Query Search Chain.

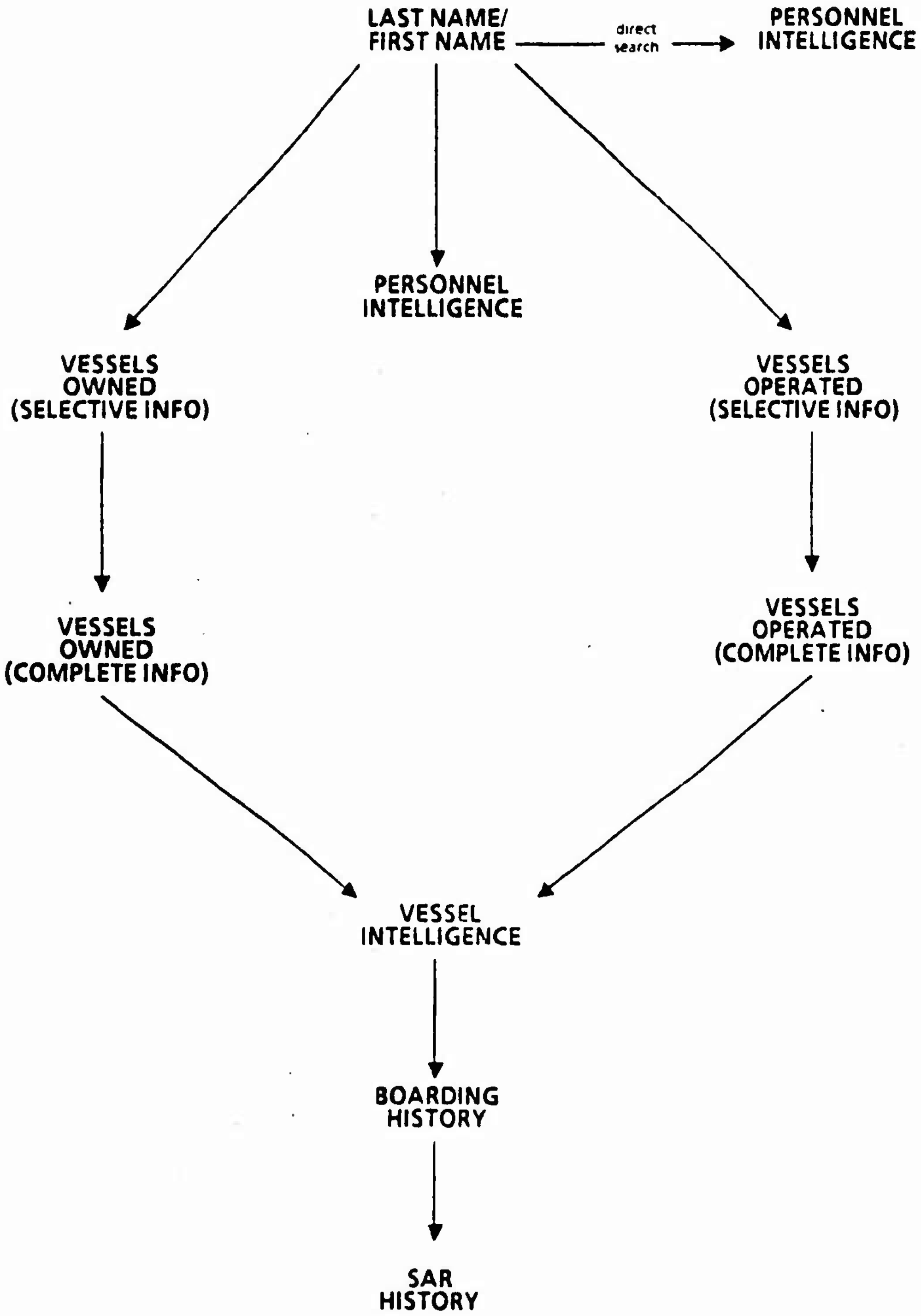
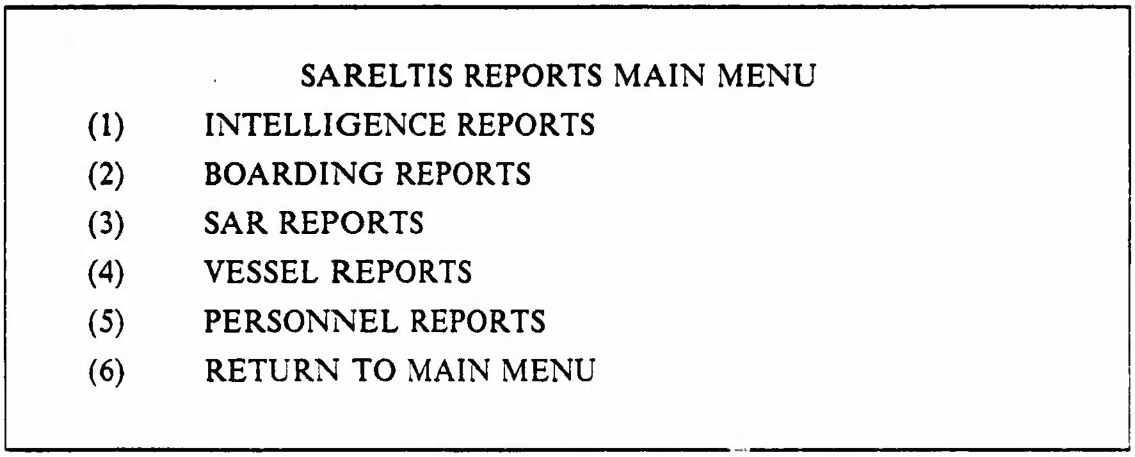

LAST NAME/ FIRST NAME
direct search
PERSONNEL INTELLIGENCE
PERSONNEL INTELLIGENCE
VESSELS OWNED (SELECTIVE INFO)
VESSELS OPERATED (SELECTIVE INFO)
VESSELS OWNED (COMPLETE INFO)
VESSELS OPERATED (COMPLETE INFO)
VESSEL INTELLIGENCE
BOARDING HISTORY
SAR HISTORY

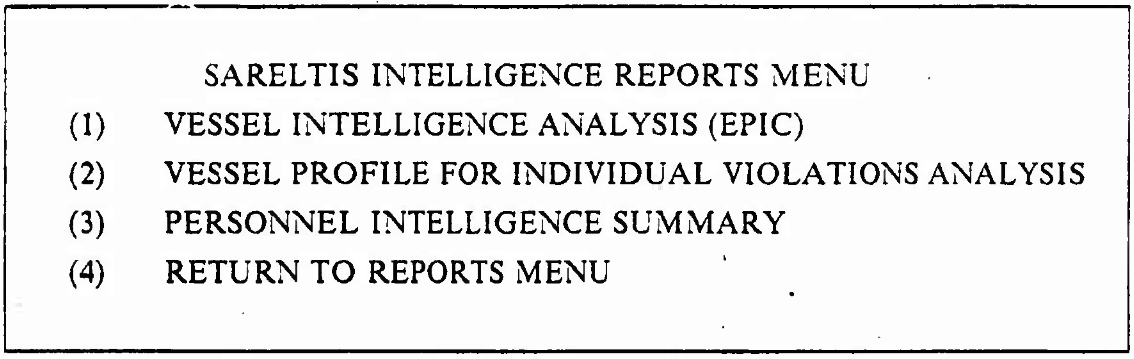
Figure 4.6 Personnel Query Search Chain.



SARELTIS REPORTS MAIN MENU

(1) INTELLIGENCE REPORTS
(2) BOARDING REPORTS
(3) SAR REPORTS
(4) VESSEL REPORTS
(5) PERSONNEL REPORTS
(6) RETURN TO MAIN MENU

Figure 4.7 SARELTIS Reports Main Menu.



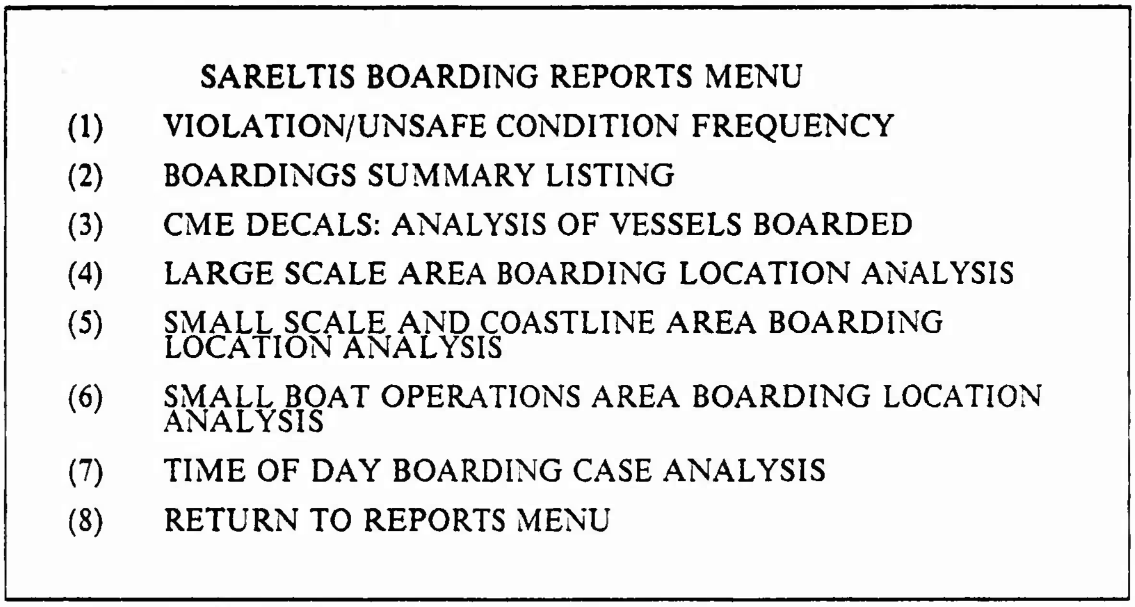
SARELTIS INTELLIGENCE REPORTS MENU

(1) VESSEL INTELLIGENCE ANALYSIS (EPIC)
(2) VESSEL PROFILE FOR INDIVIDUAL VIOLATIONS ANALYSIS
(3) PERSONNEL INTELLIGENCE SUMMARY
(4) RETURN TO REPORTS MENU

Figure 4.8 SARELTIS Intelligence Reports Menu.

f. SARELTIS Utilities Menu (Figure 4.13)

From the Utilities Menu, the user can initialize SARELTIS by selecting option (1). This program is automatically invoked at the start of any SARELTIS session and is also executed upon exiting the system. Option (2) will pack or compress the database to more efficiently use the existing limited hard disk space. It recommended that the user perform a pack of the database at least once per month. Prior to executing a pack of the database, the user should make a backup copy of the SARELTIS database in the event that an unanticipated error should occur during the packing process. Option (3) will display the procedures for backing up the database. In order to backup the database, the user must exit from SARELTIS (via option (7) on the MASTER MENU) and execute a DOS BATCH File located on the hard disk



SARELTIS BOARDING REPORTS MENU

(1) VIOLATION/UNSAFE CONDITION FREQUENCY

(2) BOARDINGS SUMMARY LISTING

(3) CME DECALS: ANALYSIS OF VESSELS BOARDED

(4) LARGE SCALE AREA BOARDING LOCATION ANALYSIS

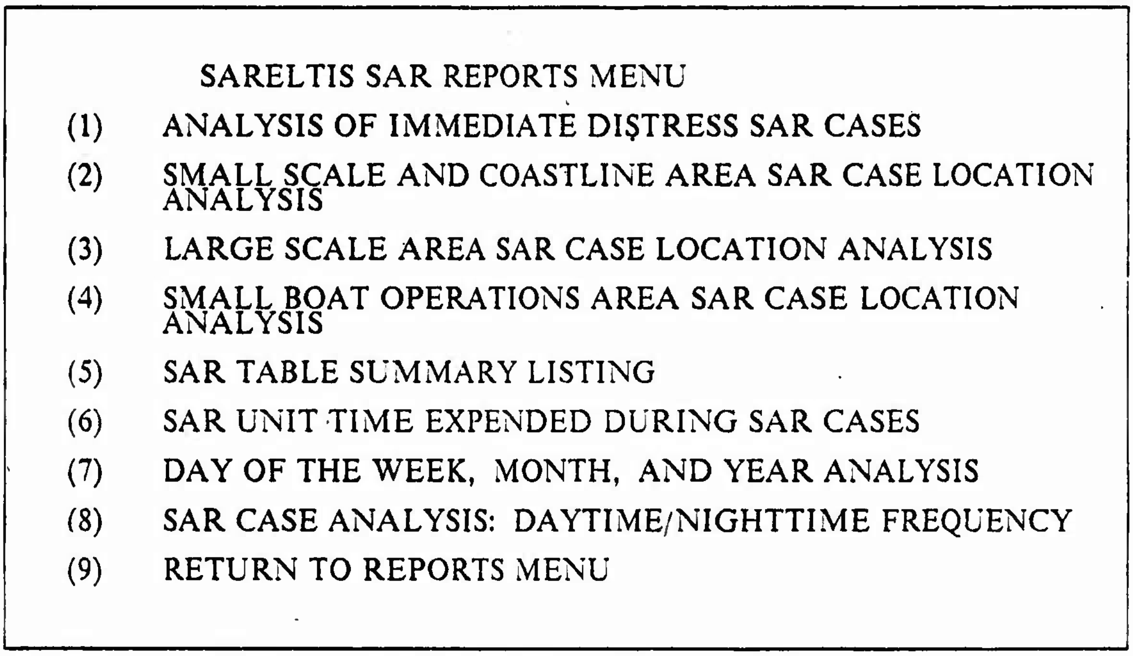
(5) SMALL SCALE AND COASTLINE AREA BOARDING LOCATION ANALYSIS

(6) SMALL BOAT OPERATIONS AREA BOARDING LOCATION ANALYSIS

(7) TIME OF DAY BOARDING CASE ANALYSIS

(8) RETURN TO REPORTS MENU

Figure 4.9 SARELTIS Boarding Reports Menu.



SARELTIS SAR REPORTS MENU

(1) ANALYSIS OF IMMEDIATE DISTRESS SAR CASES

(2) SMALL SCALE AND COASTLINE AREA SAR CASE LOCATION ANALYSIS

(3) LARGE SCALE AREA SAR CASE LOCATION ANALYSIS

(4) SMALL BOAT OPERATIONS AREA SAR CASE LOCATION ANALYSIS

(5) SAR TABLE SUMMARY LISTING

(6) SAR UNIT TIME EXPENDED DURING SAR CASES

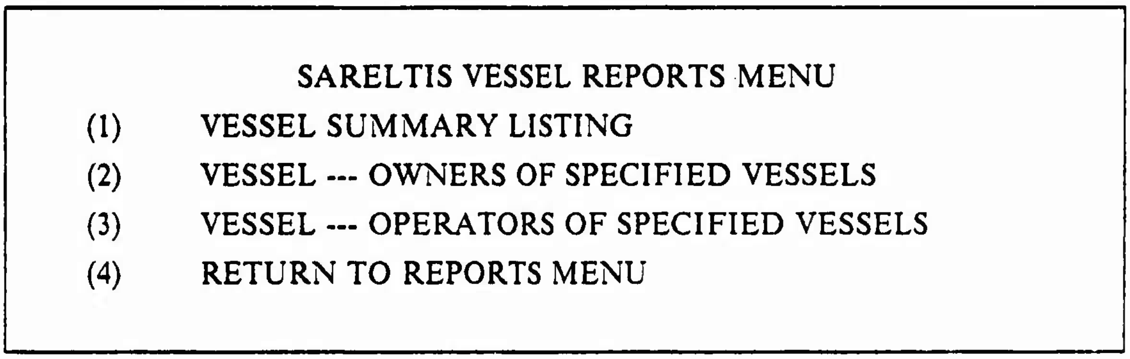
(7) DAY OF THE WEEK, MONTH, AND YEAR ANALYSIS

(8) SAR CASE ANALYSIS: DAYTIME/NIGHTTIME FREQUENCY

(9) RETURN TO REPORTS MENU

Figure 4.10 SARELTIS SAR Reports Menu.

called "BACKDBMS.BAT". Since this backup will take a minimum of six floppy disks, the user should ensure that at least six floppy disks have been formatted prior to executing this program. To format a floppy disk, type the command "PATH UTIL"



SARELTIS VESSEL REPORTS MENU

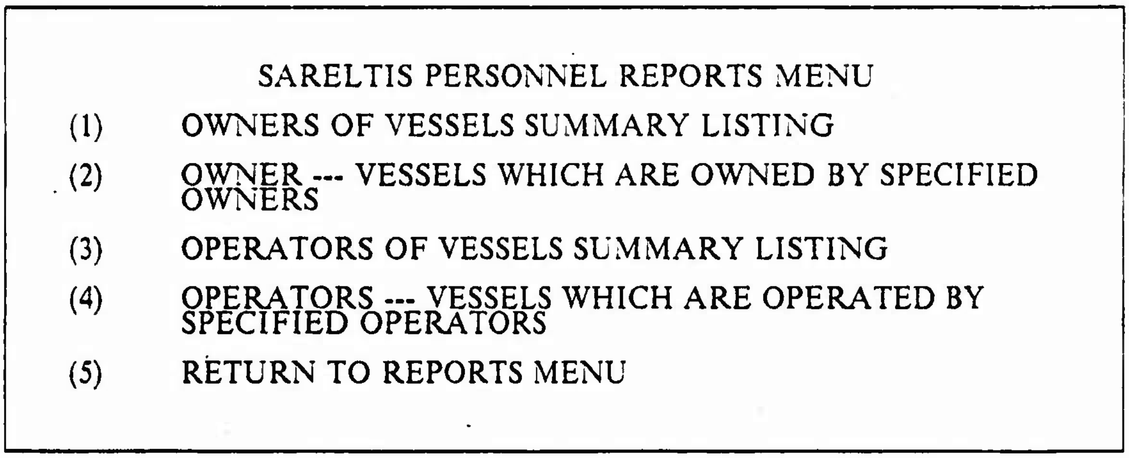
(1) VESSEL SUMMARY LISTING

(2) VESSEL --- OWNERS OF SPECIFIED VESSELS

(3) VESSEL --- OPERATORS OF SPECIFIED VESSELS

(4) RETURN TO REPORTS MENU

Figure 4.11 SARELTIS Vessel Reports Menu.



SARELTIS PERSONNEL REPORTS MENU

(1) OWNERS OF VESSELS SUMMARY LISTING

(2) OWNER --- VESSELS WHICH ARE OWNED BY SPECIFIED OWNERS

(3) OPERATORS OF VESSELS SUMMARY LISTING

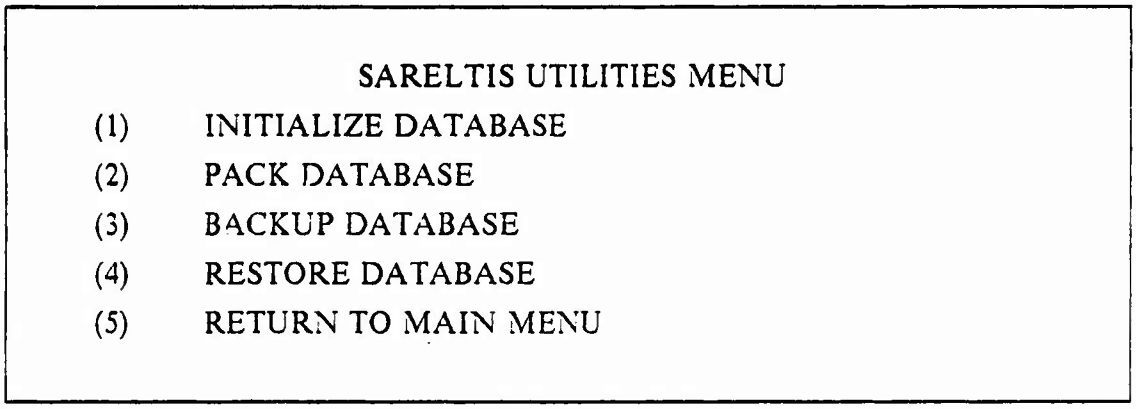
(4) OPERATORS --- VESSELS WHICH ARE OPERATED BY SPECIFIED OPERATORS

(5) RETURN TO REPORTS MENU

Figure 4.12 SARELTIS Personnel Reports Menu.

at the system prompt "C:>", then type"FORMAT A:" at the next system prompt "C:>". The system will prompt the user to insert the disks to be formatted into floppy drive A: and will indicate when the process has been completed. After ensuring that the user has prepared enough formatted floppy disks, the user is ready to execute the batch file. The user should type "BACKDBMS" at the system prompt "C:>" and press the <ENTER> key to execute the program. Follow the system prompts to remove and insert floppy disks. When the database has been backed up the program will automatically return the user to SARELTIS. In the unlikely event that the database has "crashed" (e.g. nothing seems to work as advertised), the user may recover the database from the last back-up. Similar to the back up process, option (4) details the specific procedures in order to recover the database. Basically, the user must exit SARELTIS and at the system prompt type "RESTDBMS" and press the

<ENTER> key. This executes the DOS BATCH File "RESTDBMS.BAT". The user should follow any instructions that the computer displays on the screen. When the database has been fully recovered, the program will automatically return the user to the SARELTIS system.



```
                SARELTIS UTILITIES MENU
(1)     INITIALIZE DATABASE
(2)     PACK DATABASE
(3)     BACKUP DATABASE
(4)     RESTORE DATABASE
(5)     RETURN TO MAIN MENU
```

Figure 4.13 SARELTIS Utilities Menu.

E. EXITING SARELTIS

To exit SARELTIS and return to the computer's operating system, select option (7) from the SARELTIS Main Menu. This will automatically close all open files and remove any temporary relations/tables and variables that have been created during the operation of the system. The user should not exit the system by merely turning the computer off. Data could be lost or incorrectly entered upon losing system power. Once SARELTIS has completed closing all files and removing temporary tables and variables, it will return the user to the system prompt: "C:>". Once the system prompt is displayed and the computer is no longer needed, the power may safely be turned off without danger to the SARELTIS database.

F. SARELTIS MAINTENANCE

Maintenance to the SARELTIS application programs should not be undertaken by the casual user. Before performing any alterations to the source code, the user should invest considerable time and effort in understanding the proper techniques of software programming and documentation.

Due to the high degree of interdependencies between the program modules, the user must be aware that a seemingly minor alteration could have far-reaching unseen effects. The system may appear to operate correctly, but in reality, the data being stored, modified, or printed could be inaccurate or even lost.

Users desiring to modify the source code should have a thorough knowledge of relational database structures, structured programming techniques, and the R:base 5000 command language. Prior to making any changes the user should ensure that the database is backed up and that copies of the applications programs have been made in the event that the modifications do not function properly. Above all, the user should NEVER attempt to make modifications to the source code without outside expert consultation and assistance.

Due to the size and complexity of SARELTIS, the authors recognize that errors will be discovered. This will require that the source code or database structure will need to be modified. A knowledgeable user can use the source code listings found in Appendices E through P to assist in the understanding and modification of the program modules.

V. CONCLUSIONS AND RECOMMENDATIONS

A. SARELTIS RESULTS

SARELTIS was originally conceived to merely replace Group Monterey's corporate knowledge base and provide access to historical information to assist boarding officers and Search and Rescue Coordinators. What has evolved is a system with far more capabilities than originally designed. The concept of providing only summarized frequency data was transformed to providing the capability to perform limited decision support functions. The system still produces the required standard statistical reports but also provides vessel profiles of potential violators. These profiles examine individual violation codes and search the historical boarding information to determine by violation the vessel type, use, hull material, propulsion, engine compartment type, fuel compartment, length. and year of construction of vessels found in violation of the specific code.

The system also provides SAR and ELT planning by examining "high incident times" and "high incident areas". By reviewing SAR historical information, SAR Coordinators can better plan for crew augmentation or additional floating or land resources. Additionally, by examining where SAR cases and ELT missions have occurred, Group Monterey can better plan where to strategically locate or send their floating units.

As discussed in Chapter II, the Coast Guard has been brought under scrutiny for potential harassment of the boating public by repetitively boarding the same vessel during a very small time interval. SARELTIS will provide the means to ascertain whether a particular vessel has been boarded in the operational area, when the vessel was lasted boarded, and what violations, if any, were discovered.

One of the most important aspects of SARELTIS is the capability of providing real-time information and intelligence to boarding officers and SAR Coordinators. With timely information, boarding officers can better understand and recognize potential dangerous situations prior to going on board a vessel.

B. CONCLUSIONS

This thesis was the first step toward providing better and more timely information to Coast Guard operational units. Although SARELTIS has not been put

APPENDIX A
LOGICAL MODELS

A Data Flow Diagram (DFD) is a method to graphically represent the logical functions of a system. Through the use of this very simple modeling technique, the analyst and user can examine the functions and interfaces of the existing system and easily define the components required within the proposed new system. These diagrams serve to summarize the processes and the flow of data within the logical model of the current methodology and to depict the functions of the proposed system independent of physical implementation details.

Within the diagrams, processes or activities are depicted as uniquely identified circles, and the flow of data is represented by arrows. Rectangles indicate a source or destination for some type of information or product. The number included within a process circle uniquely identifies that process and serves as a means for indicating the subprocesses of a higher level process. Higher level processes have single digit numbers, while a subprocess number consists of the parent number(s) and the subprocess number. This numbering convention allows the subfunctions of a higher level function to be easily identified. [Ref. 2: p. 282].

A second tool that is very useful for the logical modeling of a database system is the Bachman diagram. This diagram depicts the entities that are present within the system, represented as boxes, and their interrelationships, which are depicted by arrows between the entities. Representing the logical structure of the proposed database, this diagram sets a foundation for the development of the physical database structure [Ref. 3: p. 39].

Within this appendix, the existing search and rescue and law enforcement system processes are depicted by Figures A.1 through A.4, the proposed new systems as Figures A.5 through A.10, and the combined database logical structure as Figure A.11.

into production at Group Monterey, it will be implemented there in September 1987. It is anticipated that its use will stimulate further interest in the SARELTIS concept and create other needs not previously identiiied nor programmed in the first version.

C. SUGGESTIONS FOR FURTHER RESEARCH

This thesis successfully analyzed, designed, and implemented an automated intelligence system to assist small Coast Guard units similar to Group Monterey. The robustness of the system has been demonstrated in its preparation of over 45 standard reports for use by the small unit to assist in recordkeeping, intelligence reporting, and resource planning. Although not originally conceived, a limited decision support capability was built into the system. This seems to be one of the potential areas for further research. The reports designed in SARELTIS are by no means complete. It is envisioned that with the proper tools the SARELTIS database could be used as the basis for a comprehensive decision support system.

Another area of potential research that was beyond the scope of this thesis was the interfacing to a central database. By telecommunicating the data collected by the system to a Coast Guard district office or area office, the SARELTIS concept can be transported to a larger geographic area. Terminals could be connected via MODEM to this online database to provide the most up-to-date information possible.

Data entry into the system could also be improved. If SARELTIS could automatically receive intelligence information from local or federal law enforcement agencies, this would eliminate the necessity to manually enter the information via the computer keyboard. Additionally, if an optical mark boarding form was used by boarding officers, manual data entry could also be reduced with the aid of an optical mark reader.

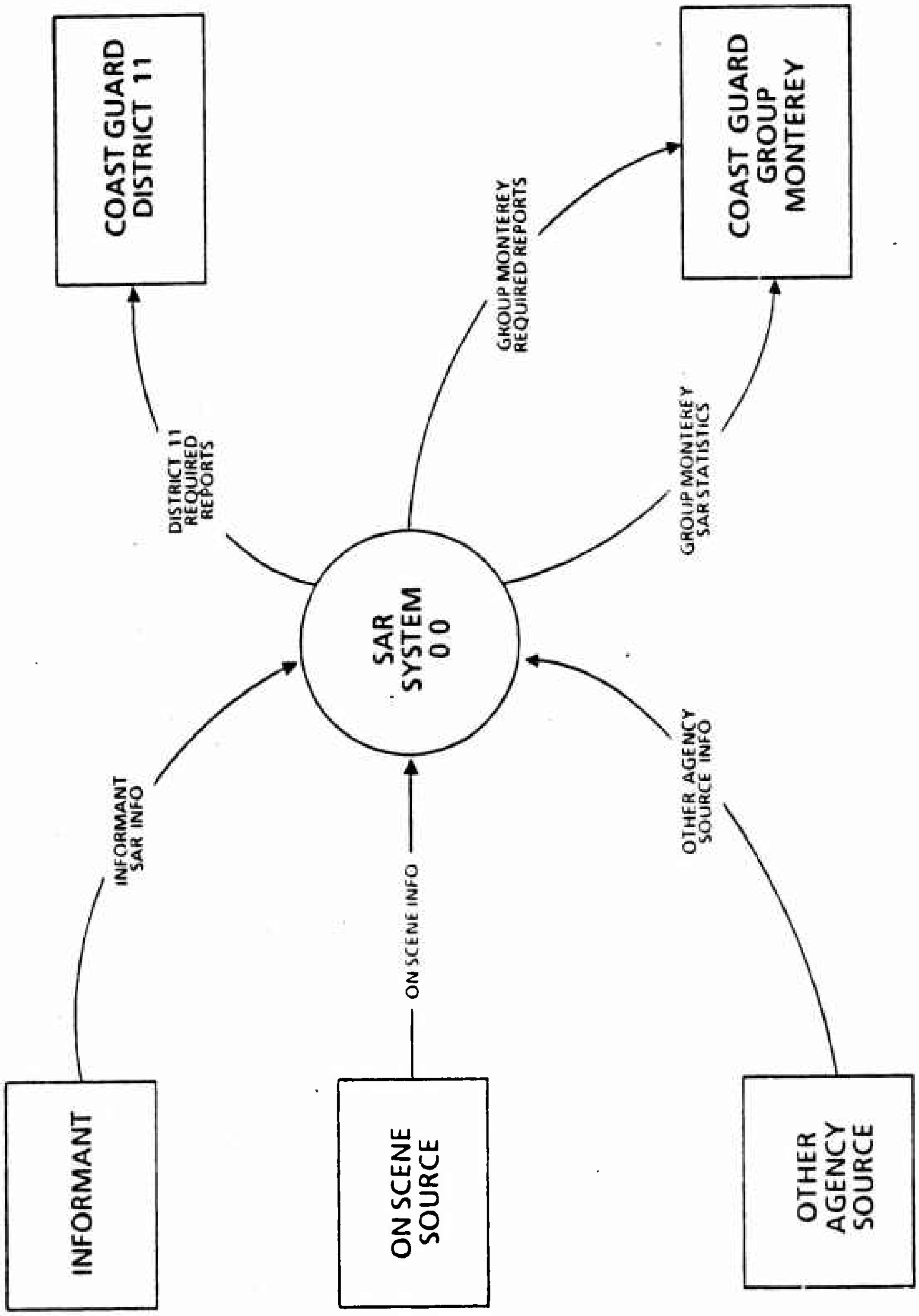


Figure A.1 Existing SAR System - Level 0.

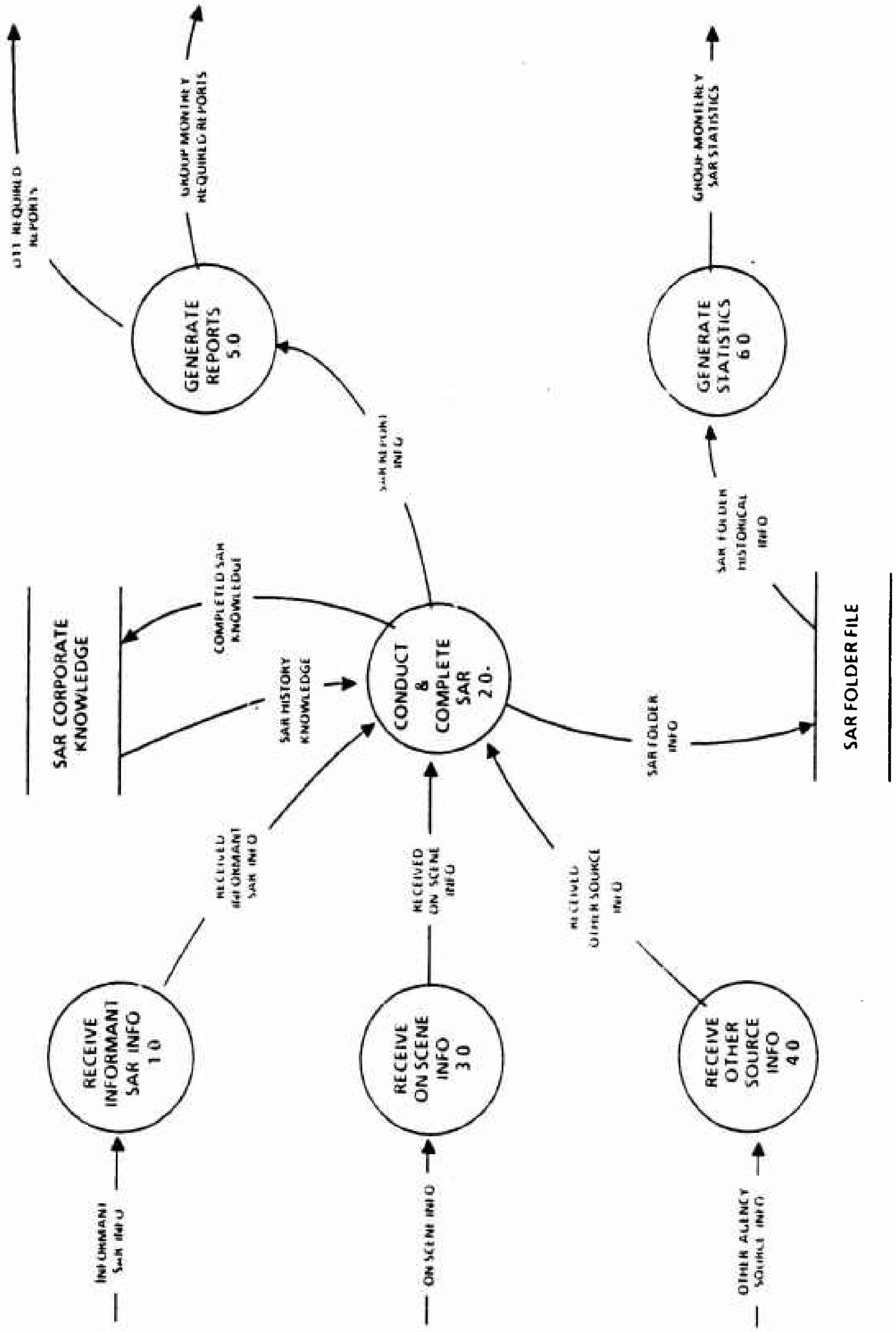


Figure A.2 Existing SAR System - Level 1.

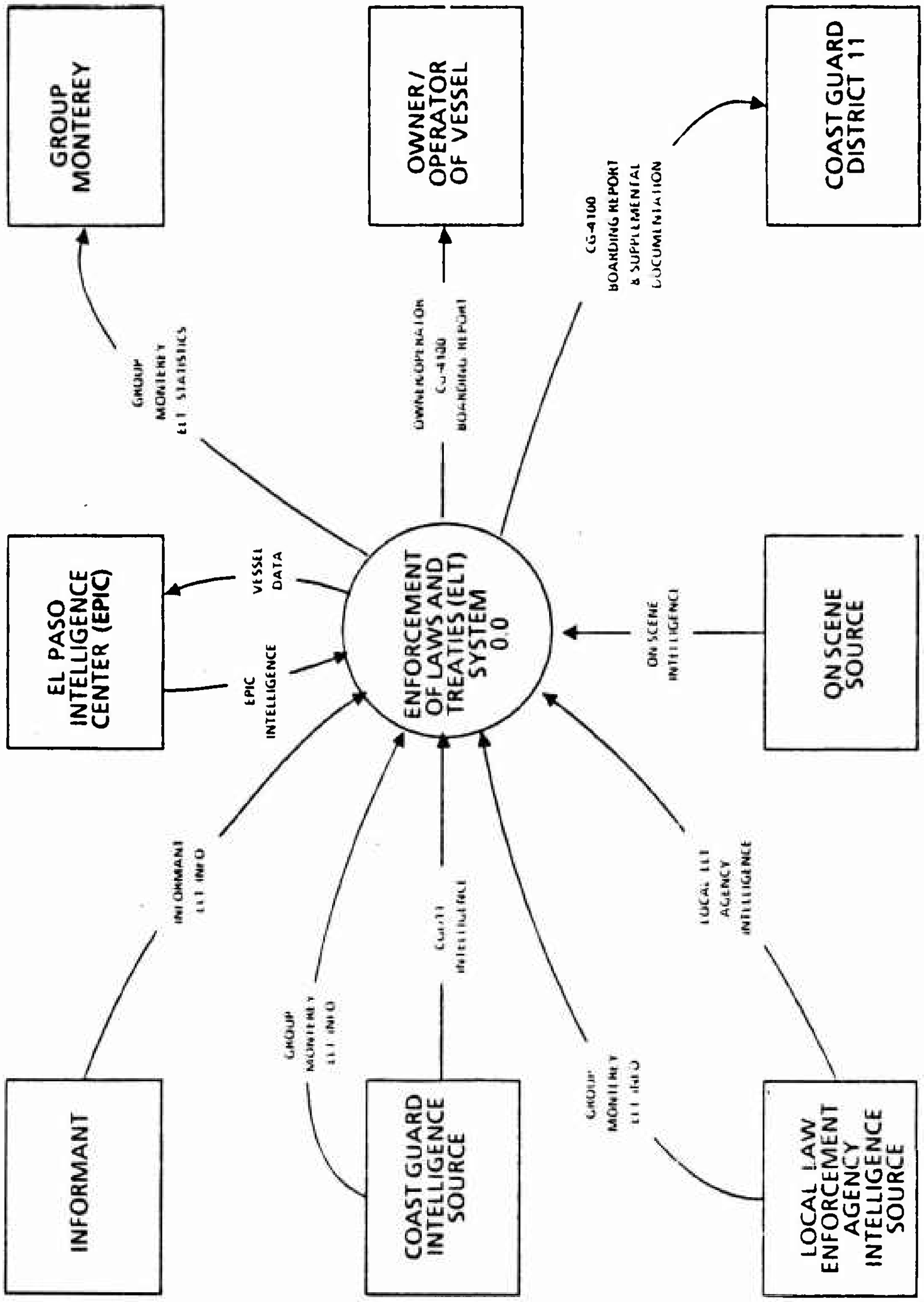

GROUP MONTEREY
OWNER / OPERATOR OF VESSEL
COAST GUARD DISTRICT 11
EL PASO INTELLIGENCE CENTER (EPIC)
ENFORCEMENT OF LAWS AND TREATIES (ELT) SYSTEM 0.0
ON SCENE SOURCE
INFORMANT
COAST GUARD INTELLIGENCE SOURCE
LOCAL LAW ENFORCEMENT AGENCY INTELLIGENCE SOURCE
GROUP MONTEREY ELT STATISTICS
OWNER/OPERATOR CG-4100 BOARDING REPORT
CG-4100 BOARDING REPORT & SUPPLEMENTAL DOCUMENTATION
VESSEL DATA
EPIC INTELLIGENCE
ON SCENE INTELLIGENCE
INFORMANT ELT INFO
GROUP MONTEREY ELT INFO
CGITT INTELLIGENCE
GROUP MONTEREY ELT INFO
LOCAL ELT AGENCY INTELLIGENCE

Figure A.3 Existing ELT System - Level 0.

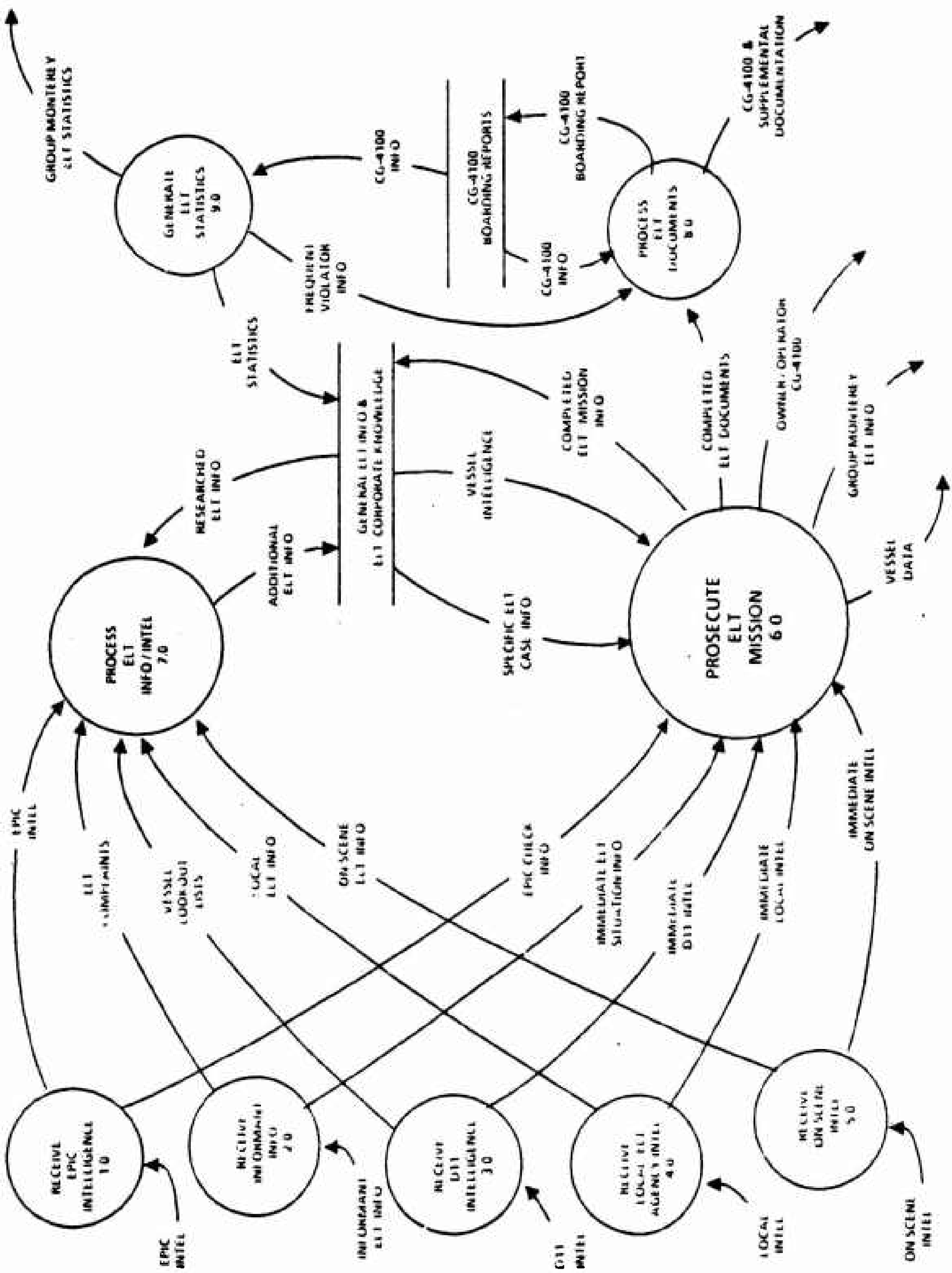


Figure A.4 Existing ELT System - Level 1.

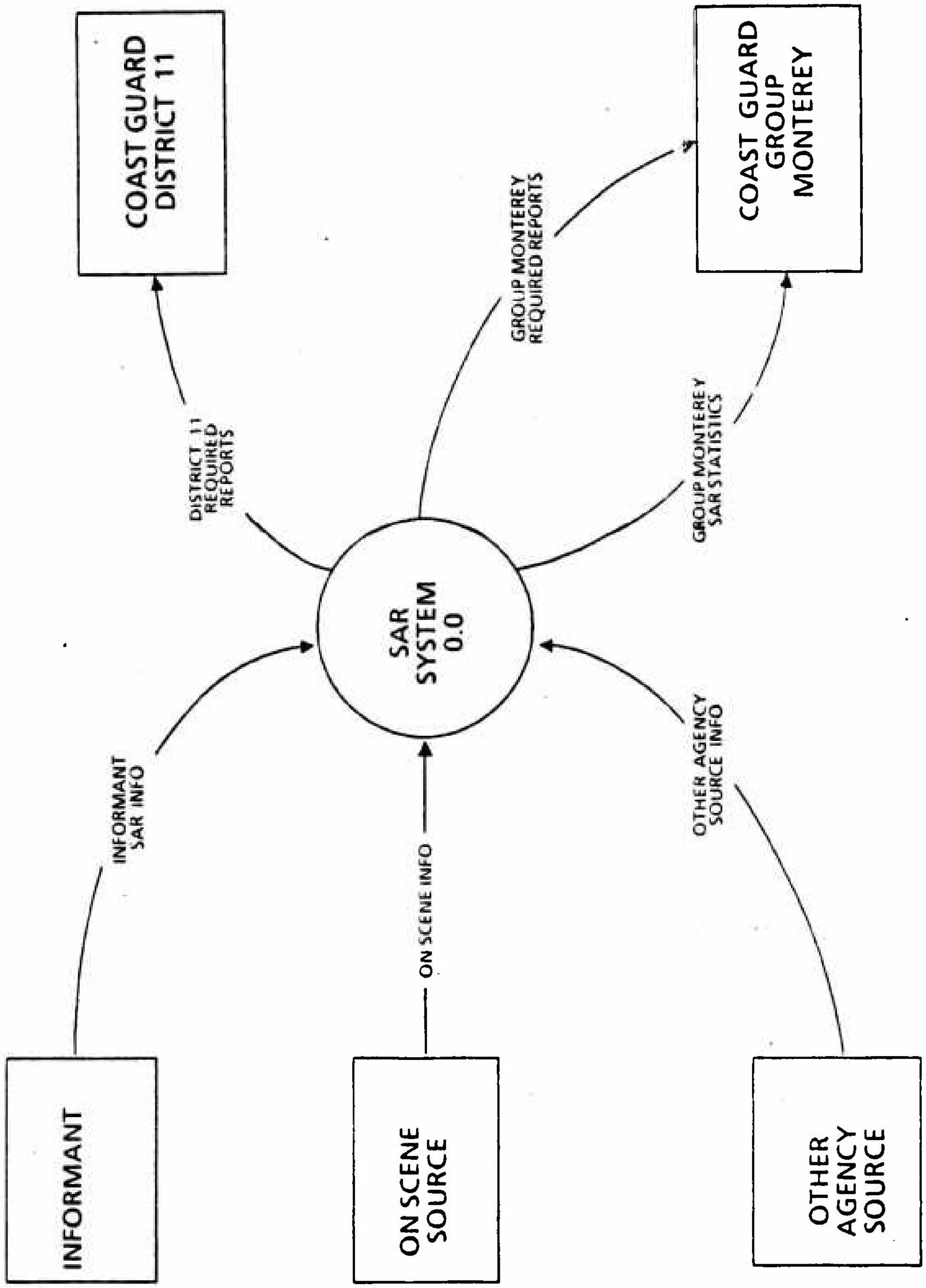


Figure A.5 Proposed SAR System - Level 0.

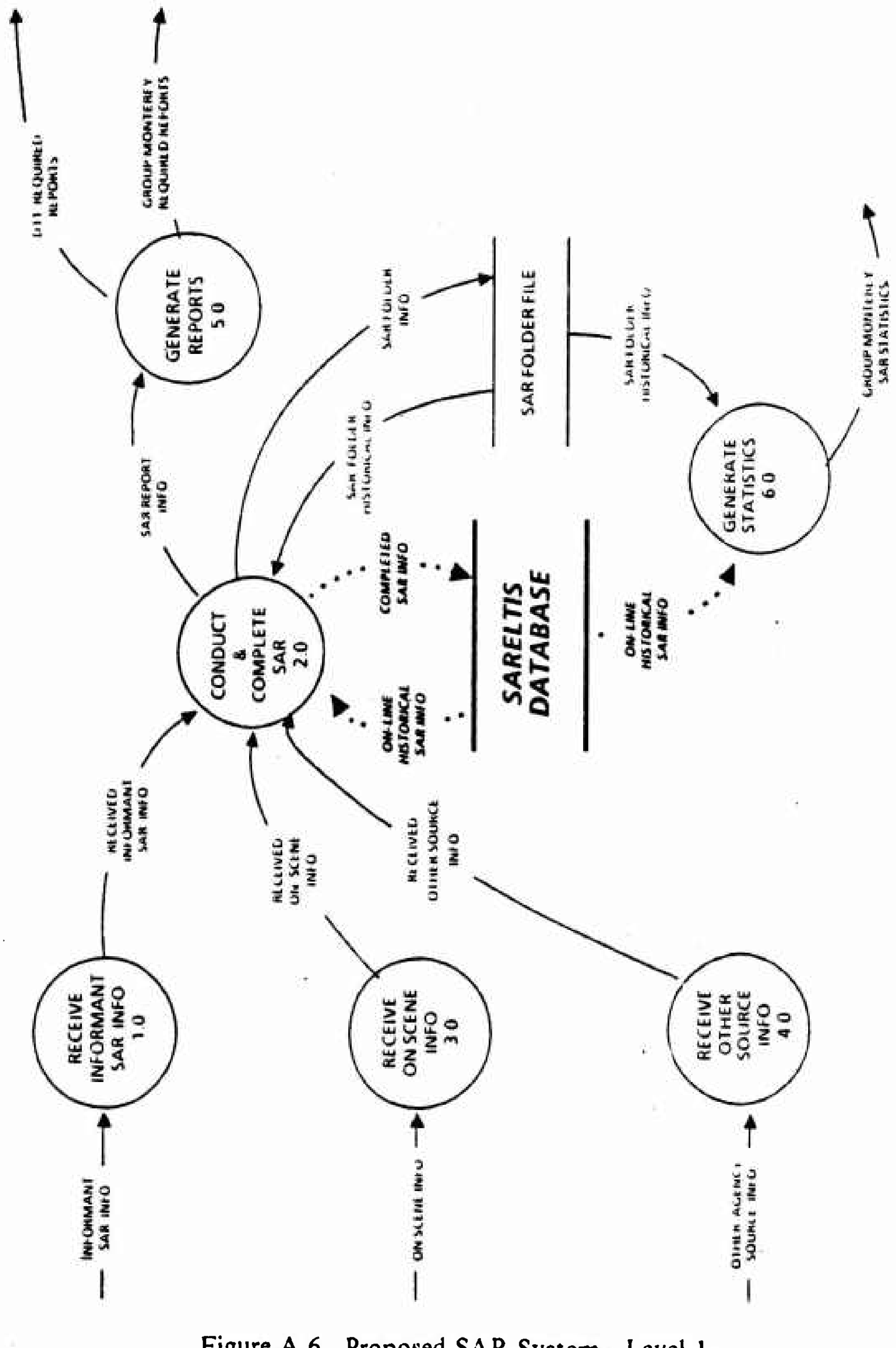

INFORMANT SAR INFO
ON SCENE INFO
OTHER AGENCY SOURCE INFO
RECEIVE INFORMANT SAR INFO 1.0
RECEIVE ON SCENE INFO 3.0
RECEIVE OTHER SOURCE INFO 4.0
RECEIVED INFORMANT SAR INFO
RECEIVED ON SCENE INFO
RECEIVED OTHER SOURCE INFO
CONDUCT & COMPLETE SAR 2.0
ON-LINE HISTORICAL SAR INFO
COMPLETED SAR INFO
SARELTIS DATABASE
ON-LINE HISTORICAL SAR INFO
SAR REPORT INFO
SAR FOLDER HISTORICAL INFO
SAR FOLDER INFO
SAR FOLDER FILE
SAR FOLDER HISTORICAL INFO
GENERATE REPORTS 5.0
D11 REQUIRED REPORTS
GROUP MONTEREY REQUIRED REPORTS
GENERATE STATISTICS 6.0
GROUP MONTEREY SAR STATISTICS

Figure A.6 Proposed SAR System - Level 1.

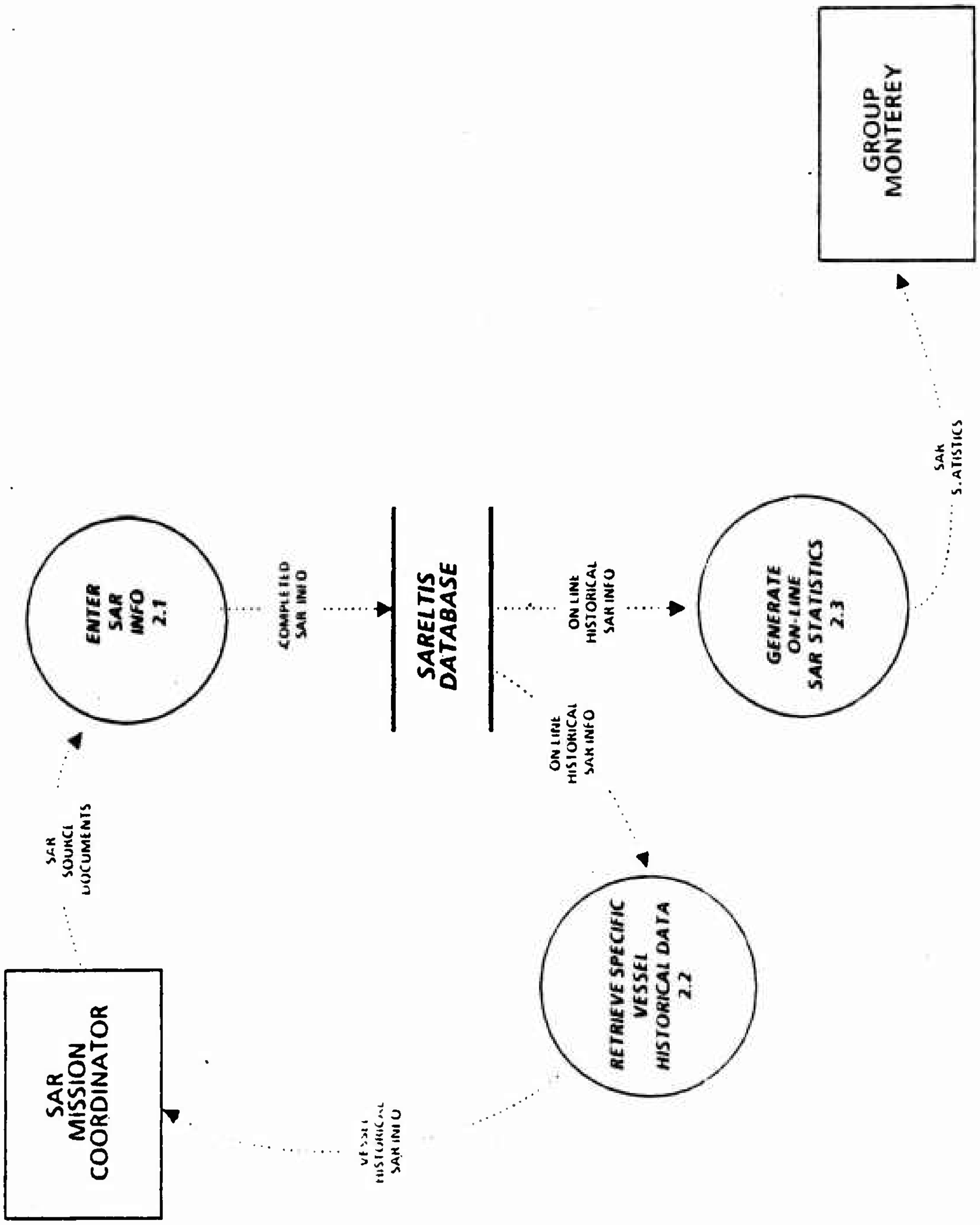

SAR MISSION COORDINATOR
SAR SOURCE DOCUMENTS
ENTER SAR INFO 2.1
COMPLETED SAR INFO
SARELTIS DATABASE
ON LINE HISTORICAL SAR INFO
GENERATE ON-LINE SAR STATISTICS 2.3
SAR STATISTICS
GROUP MONTEREY
ON LINE HISTORICAL SAR INFO
RETRIEVE SPECIFIC VESSEL HISTORICAL DATA 2.2
VESSEL HISTORICAL SAR INFO

Figure A.7 SAR System SARELTIS Automation Boundary.

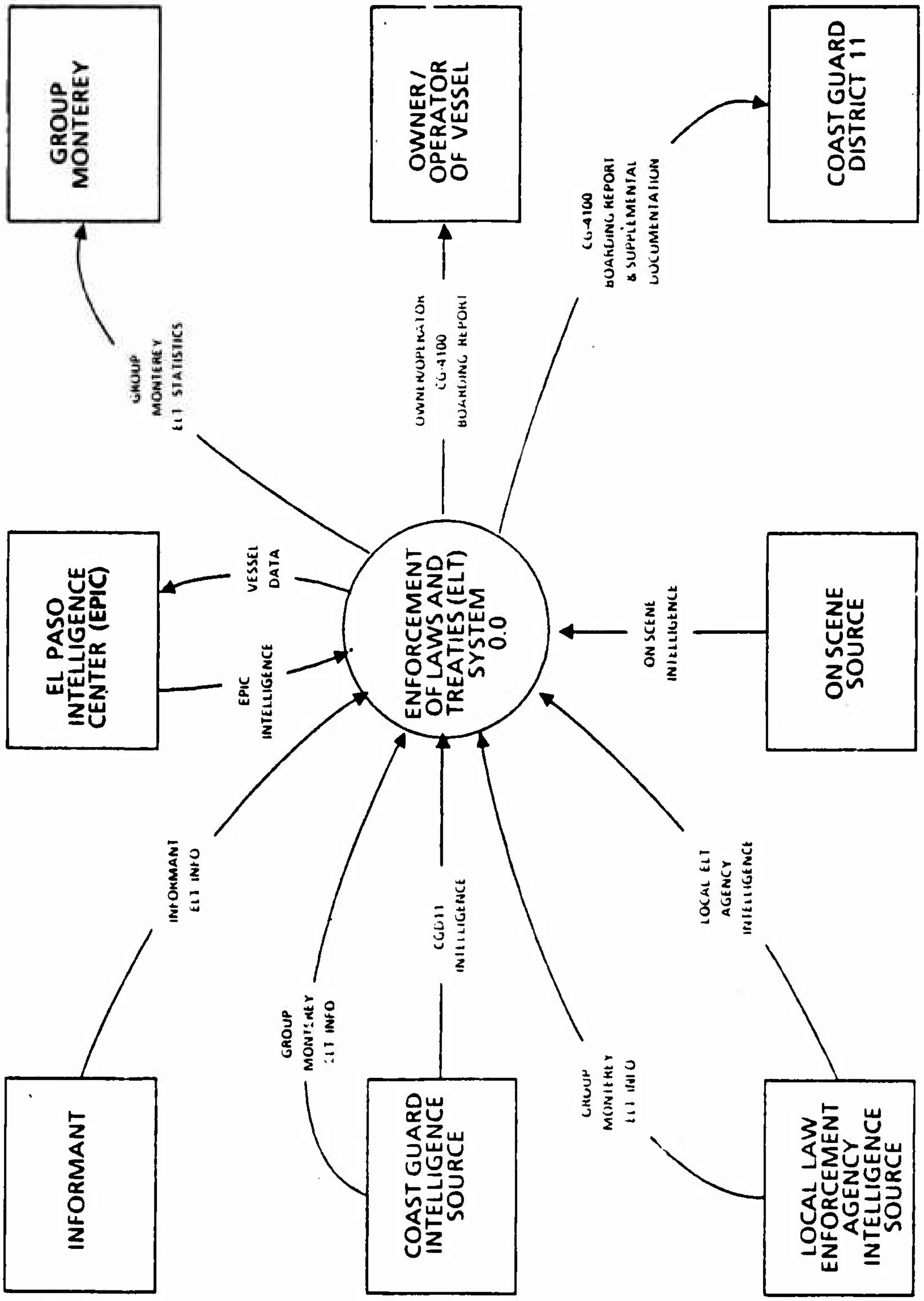

GROUP MONTEREY
OWNER / OPERATOR OF VESSEL
COAST GUARD DISTRICT 11
GROUP MONTEREY ELT STATISTICS
OWNER/OPERATOR CG-4100 BOARDING REPORT
CG-4100 BOARDING REPORT & SUPPLEMENTAL DOCUMENTATION
EL PASO INTELLIGENCE CENTER (EPIC)
VESSEL DATA
EPIC INTELLIGENCE
ENFORCEMENT OF LAWS AND TREATIES (ELT) SYSTEM 0.0
ON SCENE INTELLIGENCE
ON SCENE SOURCE
INFORMANT ELT INFO
GROUP MONTEREY ELT INFO
CGD11 INTELLIGENCE
GROUP MONTEREY ELT INFO
LOCAL ELT AGENCY INTELLIGENCE
INFORMANT
COAST GUARD INTELLIGENCE SOURCE
LOCAL LAW ENFORCEMENT AGENCY INTELLIGENCE SOURCE

Figure A.8 Proposed ELT System - Level 0.

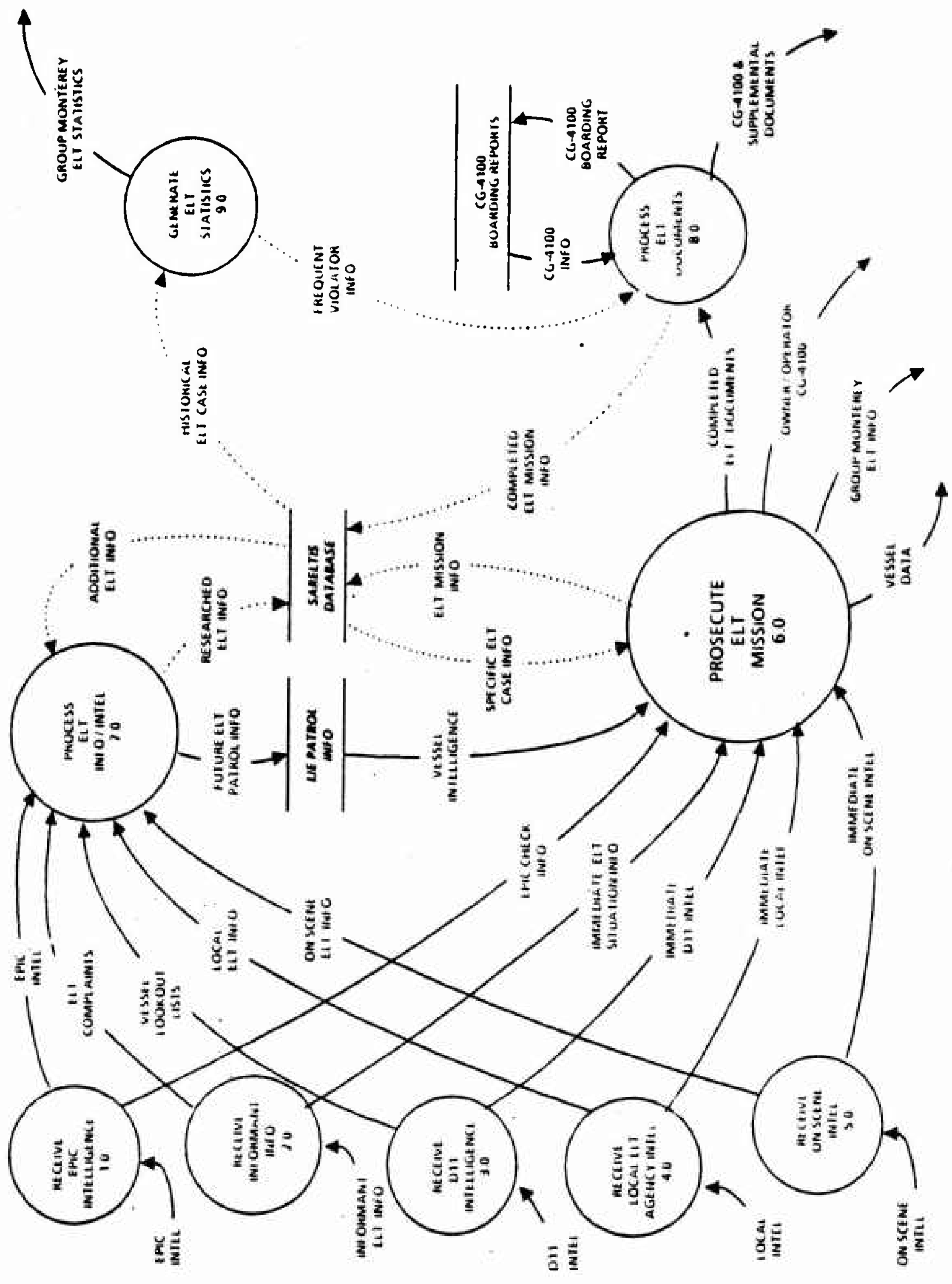


Figure A.9 Proposed ELT System - Level 1.

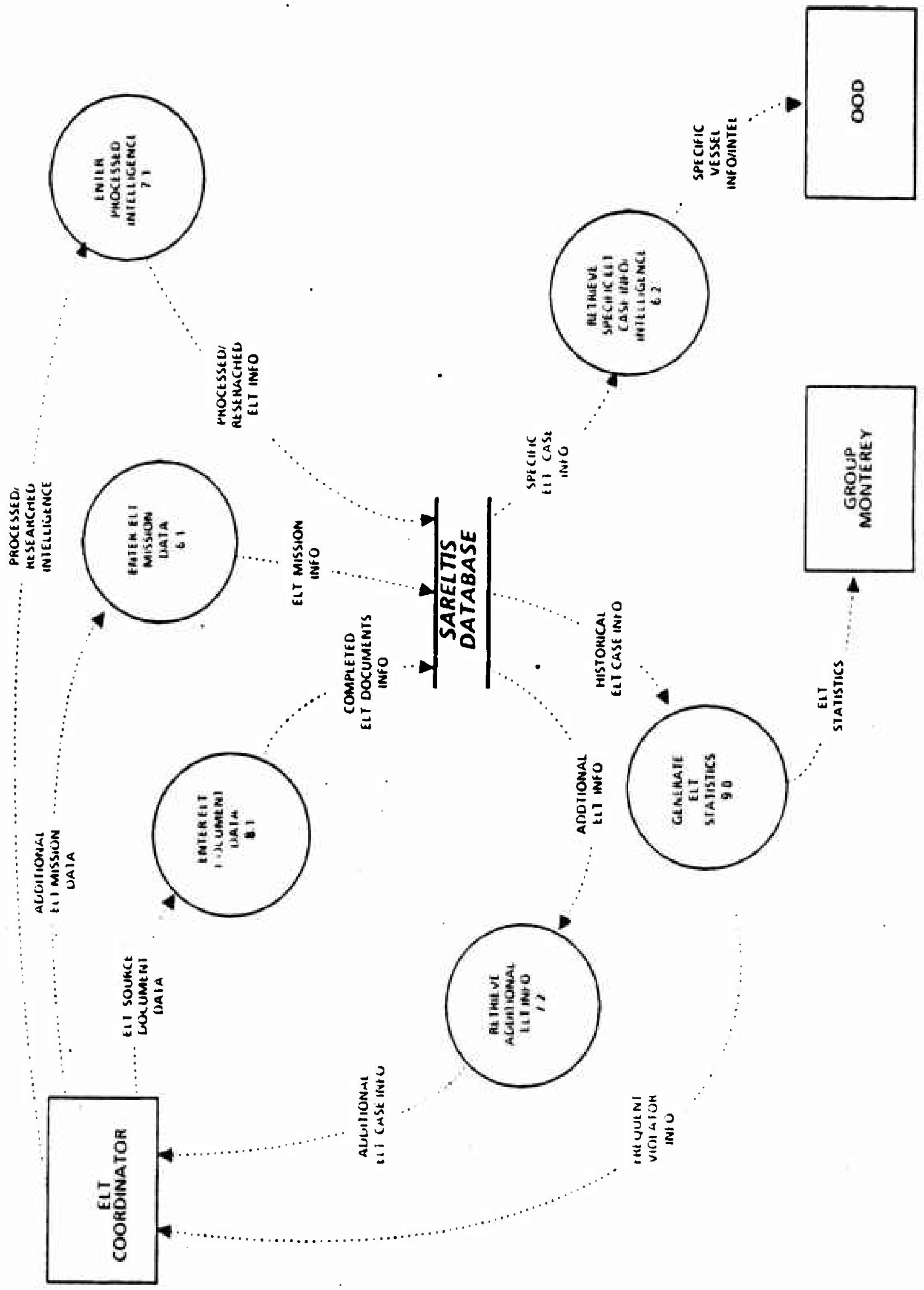

ENTER PROCESSED INTELLIGENCE 7.1
PROCESSED/ RESEARCHED INTELLIGENCE
PROCESSED/ RESEARCHED ELT INFO
ENTER ELT MISSION DATA 6.1
ELT MISSION INFO
ADDITIONAL ELT MISSION DATA
ENTER ELT DOCUMENT DATA 8.1
COMPLETED ELT DOCUMENTS INFO
ELT SOURCE DOCUMENT DATA
SARELTIS DATABASE
SPECIFIC ELT CASE INFO
RETRIEVE SPECIFIC ELT CASE INFO/ INTELLIGENCE 6.2
SPECIFIC VESSEL INFO/INTEL
OOD
HISTORICAL ELT CASE INFO
GENERATE ELT STATISTICS 9.0
ELT STATISTICS
GROUP MONTEREY
ADDIONAL ELT INFO
RETRIEVE ADDITIONAL ELT INFO 7.2
ADDITIONAL ELT CASE INFO
FREQUENT VIOLATOR INFO
ELT COORDINATOR

Figure A.10 ELT System SARELTIS Automation Boundary.

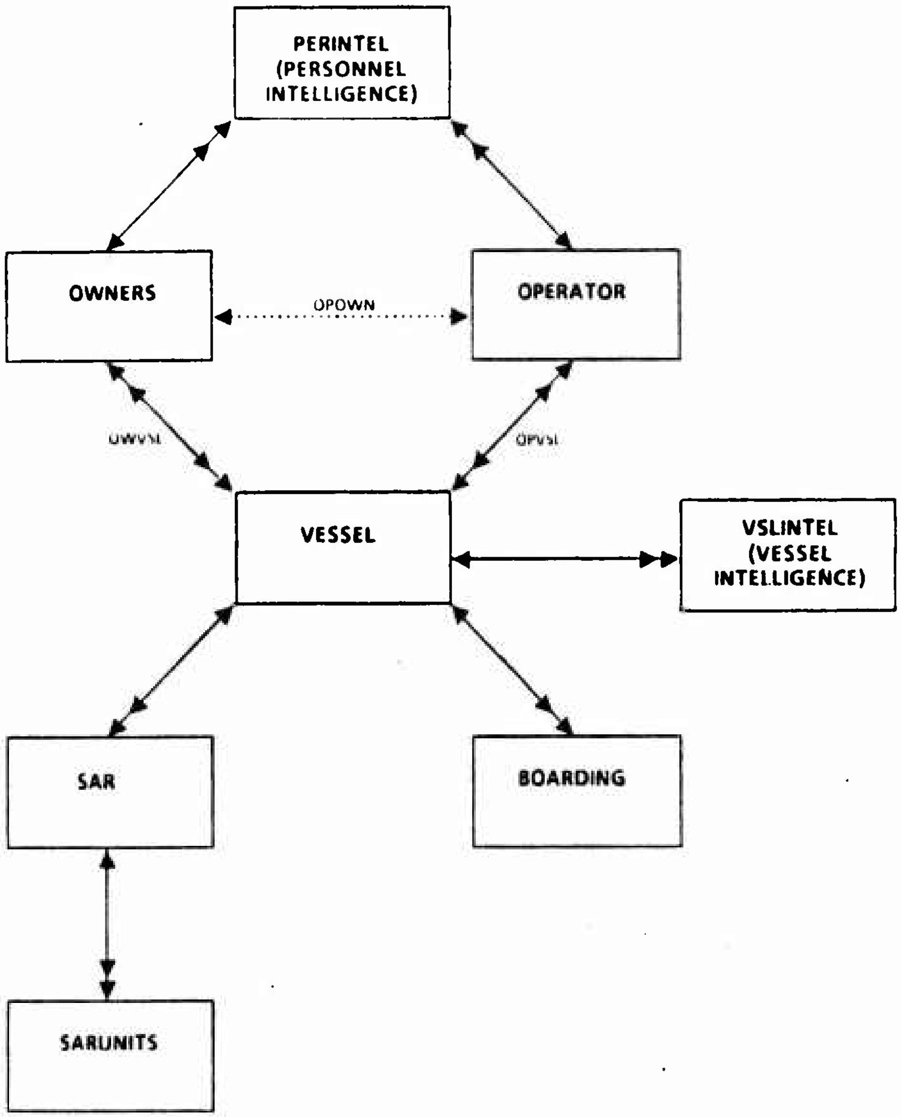
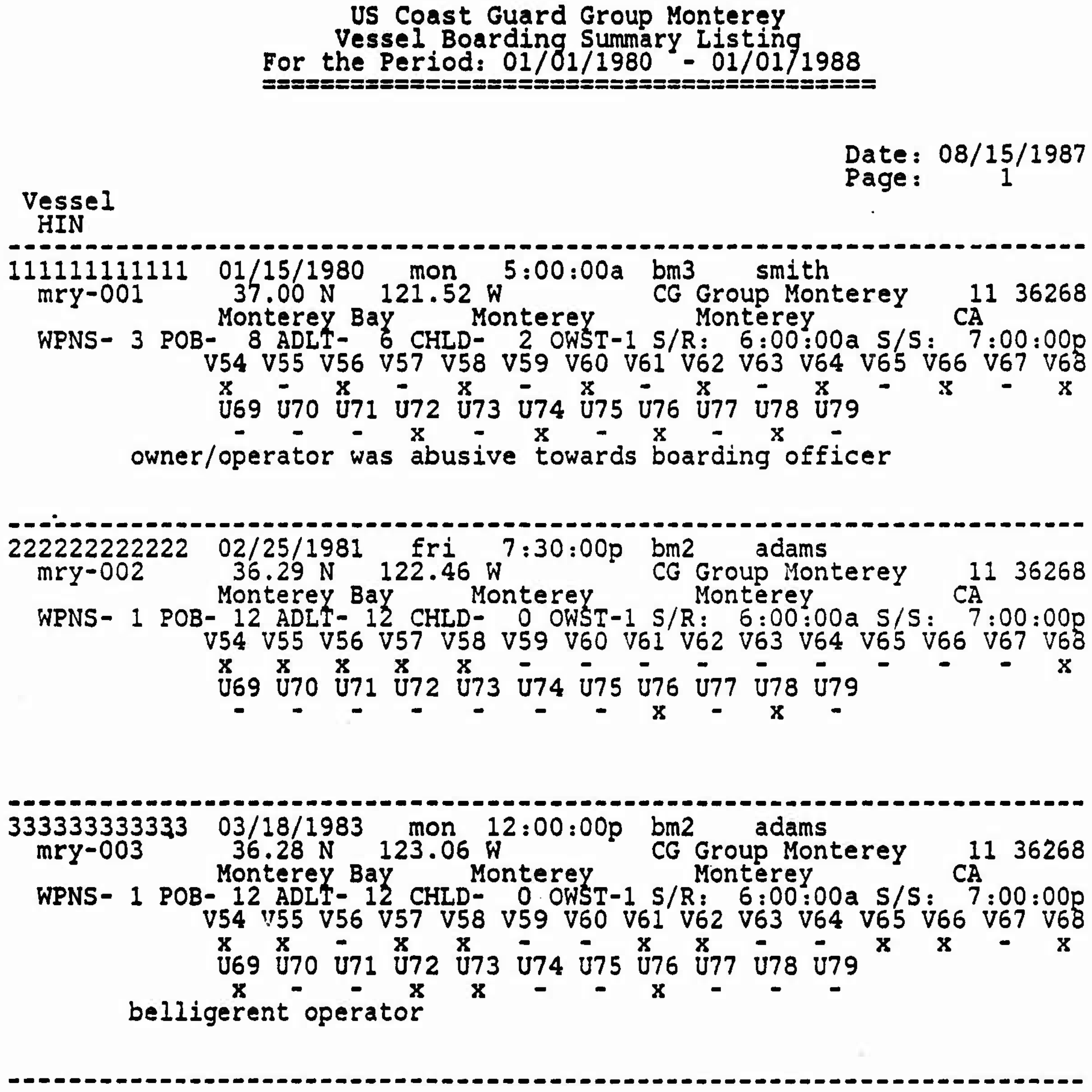

PERINTEL (PERSONNEL INTELLIGENCE)
OWNERS
OPOWN
OPERATOR
OWVSL
OPVSL
VESSEL
VSLINTEL (VESSEL INTELLIGENCE)
SAR
BOARDING
SARUNITS

Figure A.11 Bachman Diagram of Proposed System.

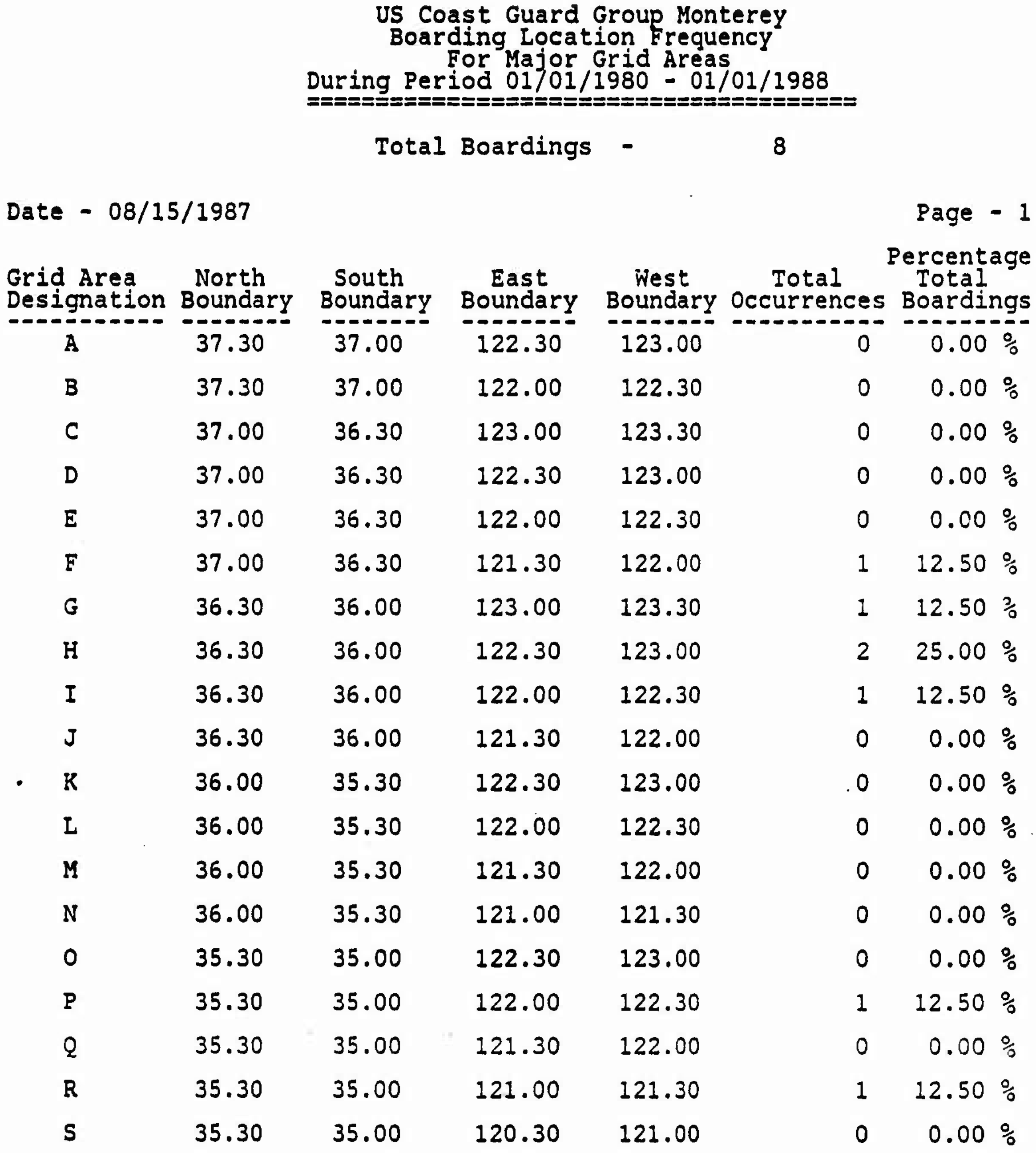
APPENDIX B
SYSTEM REPORTS

To assist Coast Guard Group Monterey with operations, planning and report generation, the SARELTIS prototype provides numerous hard copy reports. Procedures utilized to generate these reports are discussed in Chapter IV, the System User's Guide. Examples of all reports produced by the prototype are displayed in this appendix as Figures B.1 through B.45. Table 3 provides a reference for the correspondence between the figures included in this appendix and the report names called within the program listing.



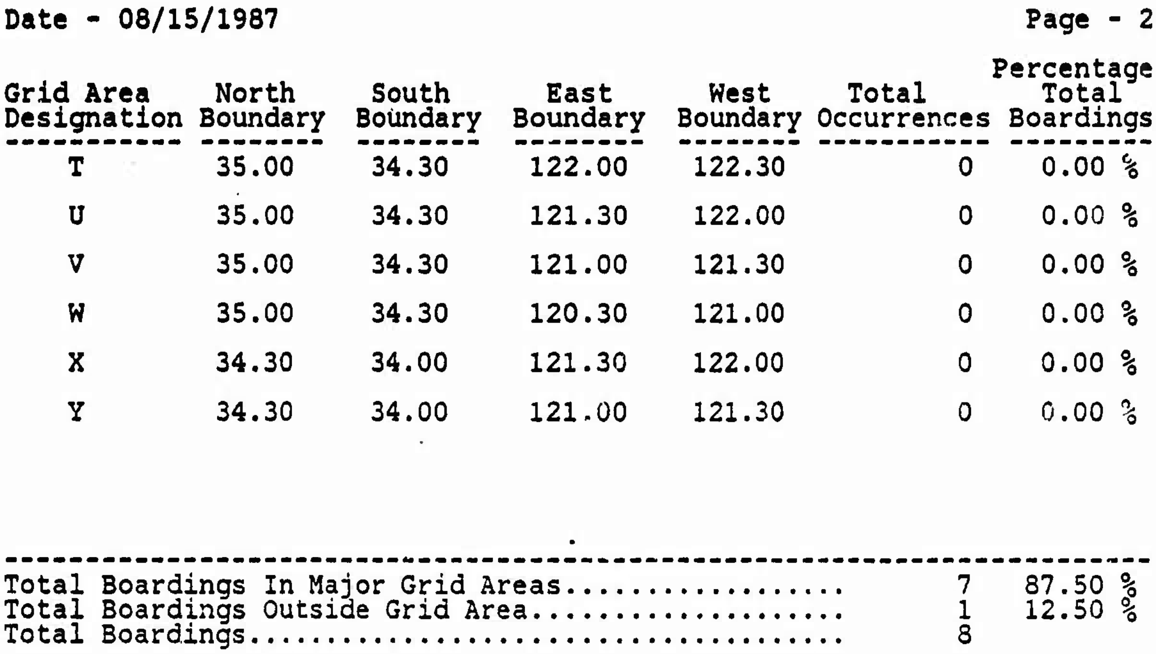
```
                       US Coast Guard Group Monterey
                      Vessel Boarding Summary Listing
                 For the Period: 01/01/1980  - 01/01/1988
                 =========================================

                                                        Date: 08/15/1987
                                                        Page:     1
 Vessel
  HIN
------------------------------------------------------------------------
111111111111  01/15/1980   mon   5:00:00a  bm3    smith
  mry-001      37.00 N   121.52 W          CG Group Monterey    11 36268
              Monterey Bay     Monterey       Monterey        CA
  WPNS- 3 POB-  8 ADLT-  6 CHLD-  2 OWST-1 S/R:  6:00:00a S/S:  7:00:00p
             V54 V55 V56 V57 V58 V59 V60 V61 V62 V63 V64 V65 V66 V67 V68
              x   -   x   -   x   -   x   -   x   -   x   -   x   -   x
              U69 U70 U71 U72 U73 U74 U75 U76 U77 U78 U79
               -   -   -   x   -   x   -   x   -   x   -
        owner/operator was abusive towards boarding officer

------------------------------------------------------------------------
222222222222  02/25/1981   fri   7:30:00p  bm2    adams
  mry-002      36.29 N   122.46 W          CG Group Monterey    11 36268
              Monterey Bay     Monterey       Monterey        CA
  WPNS- 1 POB- 12 ADLT- 12 CHLD-  0 OWST-1 S/R:  6:00:00a S/S:  7:00:00p
             V54 V55 V56 V57 V58 V59 V60 V61 V62 V63 V64 V65 V66 V67 V68
              x   x   x   x   x   -   -   -   -   -   -   -   -   -   x
              U69 U70 U71 U72 U73 U74 U75 U76 U77 U78 U79
               -   -   -   -   -   -   -   x   -   x   -


------------------------------------------------------------------------
333333333333  03/18/1983   mon  12:00:00p  bm2    adams
  mry-003      36.28 N   123.06 W          CG Group Monterey    11 36268
              Monterey Bay     Monterey       Monterey        CA
  WPNS- 1 POB- 12 ADLT- 12 CHLD-  0 OWST-1 S/R:  6:00:00a S/S:  7:00:00p
             V54 V55 V56 V57 V58 V59 V60 V61 V62 V63 V64 V65 V66 V67 V68
              x   x   -   x   x   -   -   x   x   -   -   x   x   -   x
              U69 U70 U71 U72 U73 U74 U75 U76 U77 U78 U79
               x   -   -   x   x   -   -   x   -   -   -
        belligerent operator

------------------------------------------------------------------------
```

Figure B.1 Boarding Listing.



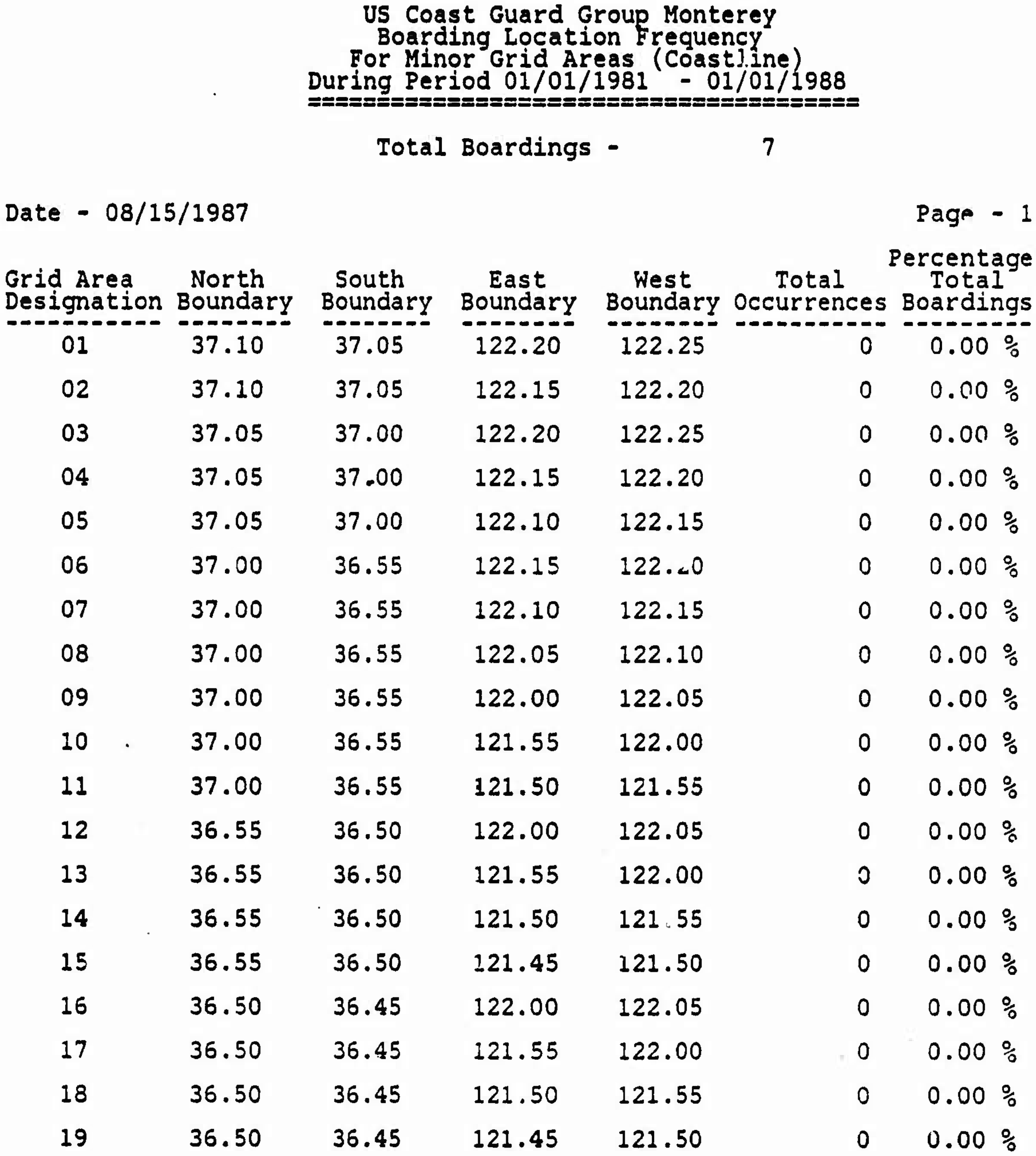
```
                          US Coast Guard Group Monterey
                           Boarding Location Frequency
                              For Major Grid Areas
                       During Period 01/01/1980 - 01/01/1988
                       ======================================

                          Total Boardings  -         8

Date - 08/15/1987                                                          Page - 1
                                                                         Percentage
Grid Area    North      South       East       West       Total        Total
Designation Boundary   Boundary   Boundary   Boundary  Occurrences  Boardings
----------- --------   --------   --------   --------  -----------  ---------
    A        37.30      37.00      122.30     123.00             0     0.00 %

    B        37.30      37.00      122.00     122.30             0     0.00 %

    C        37.00      36.30      123.00     123.30             0     0.00 %

    D        37.00      36.30      122.30     123.00             0     0.00 %

    E        37.00      36.30      122.00     122.30             0     0.00 %

    F        37.00      36.30      121.30     122.00             1    12.50 %

    G        36.30      36.00      123.00     123.30             1    12.50 %

    H        36.30      36.00      122.30     123.00             2    25.00 %

    I        36.30      36.00      122.00     122.30             1    12.50 %

    J        36.30      36.00      121.30     122.00             0     0.00 %

    K        36.00      35.30      122.30     123.00             0     0.00 %

    L        36.00      35.30      122.00     122.30             0     0.00 %

    M        36.00      35.30      121.30     122.00             0     0.00 %

    N        36.00      35.30      121.00     121.30             0     0.00 %

    O        35.30      35.00      122.30     123.00             0     0.00 %

    P        35.30      35.00      122.00     122.30             1    12.50 %

    Q        35.30      35.00      121.30     122.00             0     0.00 %

    R        35.30      35.00      121.00     121.30             1    12.50 %

    S        35.30      35.00      120.30     121.00             0     0.00 %
```

Figure B.2 Major Grid Area Boarding Frequency.



```
Date - 08/15/1987                                                    Page - 2

                                                                   Percentage
Grid Area    North      South      East       West       Total       Total
Designation Boundary   Boundary   Boundary   Boundary  Occurrences  Boardings
----------- --------   --------   --------   --------  -----------  ---------
     T       35.00      34.30     122.00     122.30             0     0.00 %

     U       35.00      34.30     121.30     122.00             0     0.00 %

     V       35.00      34.30     121.00     121.30             0     0.00 %

     W       35.00      34.30     120.30     121.00             0     0.00 %

     X       34.30      34.00     121.30     122.00             0     0.00 %

     Y       34.30      34.00     121.00     121.30             0     0.00 %




-----------------------------------------------------------------------------
Total Boardings In Major Grid Areas.................            7    87.50 %
Total Boardings Outside Grid Area...................            1    12.50 %
Total Boardings.....................................            8
```

Figure B.2 Major Grid Area Boarding Frequency. (cont'd.)



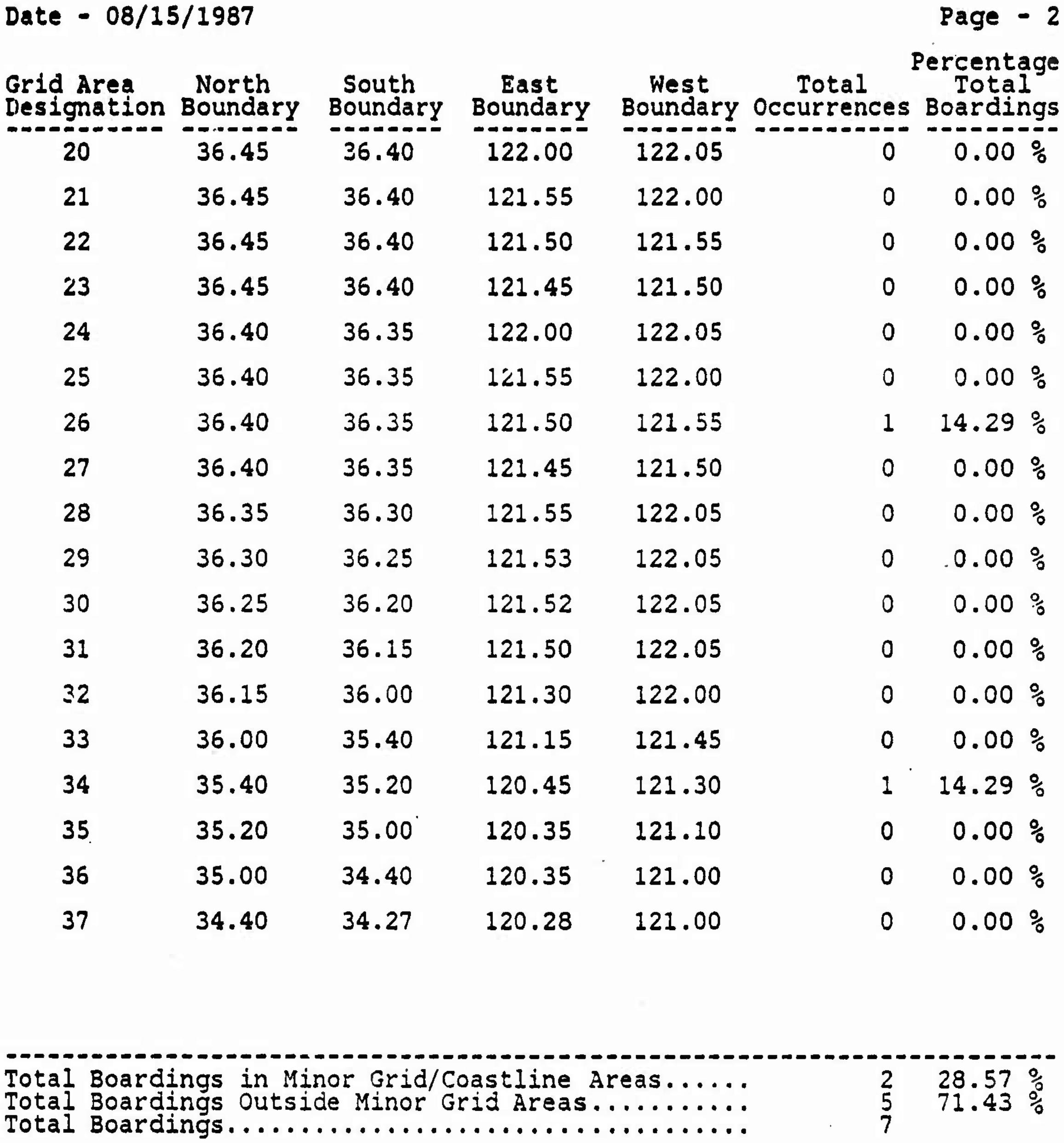
US Coast Guard Group Monterey
Boarding Location Frequency
For Minor Grid Areas (Coastline)
During Period 01/01/1981 - 01/01/1988

Total Boardings - 7

Date - 08/15/1987 Page - 1

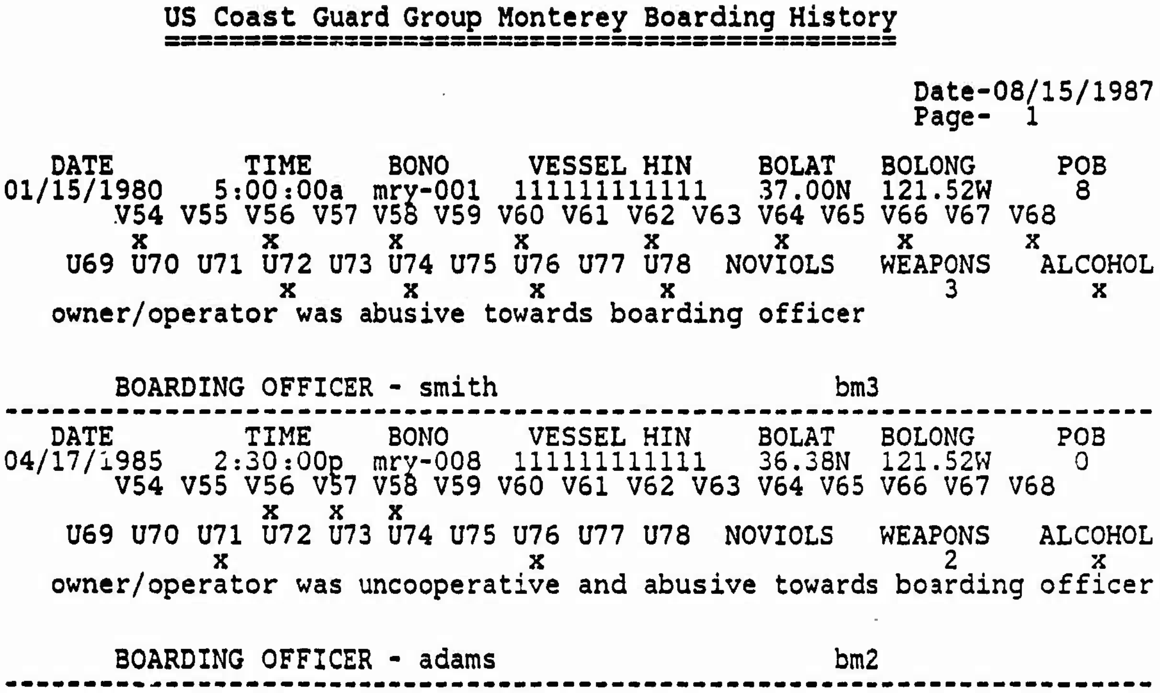
Grid Area Designation	North Boundary	South Boundary	East Boundary	West Boundary	Total Occurrences	Percentage Total Boardings
01	37.10	37.05	122.20	122.25	0	0.00 %
02	37.10	37.05	122.15	122.20	0	0.00 %
03	37.05	37.00	122.20	122.25	0	0.00 %
04	37.05	37.00	122.15	122.20	0	0.00 %
05	37.05	37.00	122.10	122.15	0	0.00 %
06	37.00	36.55	122.15	122.20	0	0.00 %
07	37.00	36.55	122.10	122.15	0	0.00 %
08	37.00	36.55	122.05	122.10	0	0.00 %
09	37.00	36.55	122.00	122.05	0	0.00 %
10	37.00	36.55	121.55	122.00	0	0.00 %
11	37.00	36.55	121.50	121.55	0	0.00 %
12	36.55	36.50	122.00	122.05	0	0.00 %
13	36.55	36.50	121.55	122.00	0	0.00 %
14	36.55	36.50	121.50	121.55	0	0.00 %
15	36.55	36.50	121.45	121.50	0	0.00 %
16	36.50	36.45	122.00	122.05	0	0.00 %
17	36.50	36.45	121.55	122.00	0	0.00 %
18	36.50	36.45	121.50	121.55	0	0.00 %
19	36.50	36.45	121.45	121.50	0	0.00 %

Figure B.3 Minor Grid Area Boarding Frequency.



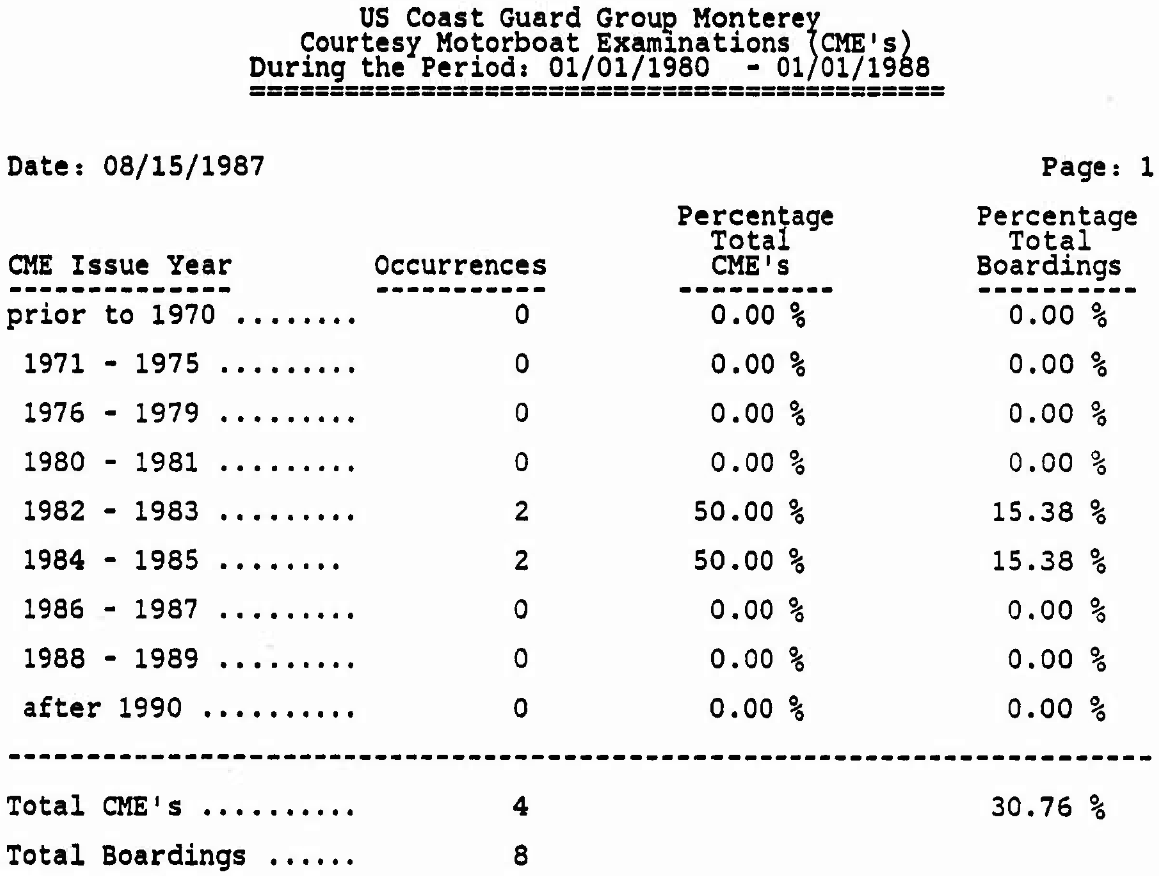
```
Date - 08/15/1987                                                    Page - 2

                                                                   Percentage
Grid Area    North      South      East       West       Total       Total
Designation Boundary   Boundary   Boundary   Boundary  Occurrences Boardings
----------- --------   --------   --------   --------  ----------- ---------
    20       36.45      36.40      122.00     122.05            0    0.00 %

    21       36.45      36.40      121.55     122.00            0    0.00 %

    22       36.45      36.40      121.50     121.55            0    0.00 %

    23       36.45      36.40      121.45     121.50            0    0.00 %

    24       36.40      36.35      122.00     122.05            0    0.00 %

    25       36.40      36.35      121.55     122.00            0    0.00 %

    26       36.40      36.35      121.50     121.55            1   14.29 %

    27       36.40      36.35      121.45     121.50            0    0.00 %

    28       36.35      36.30      121.55     122.05            0    0.00 %

    29       36.30      36.25      121.53     122.05            0    0.00 %

    30       36.25      36.20      121.52     122.05            0    0.00 %

    31       36.20      36.15      121.50     122.05            0    0.00 %

    32       36.15      36.00      121.30     122.00            0    0.00 %

    33       36.00      35.40      121.15     121.45            0    0.00 %

    34       35.40      35.20      120.45     121.30            1   14.29 %

    35       35.20      35.00      120.35     121.10            0    0.00 %

    36       35.00      34.40      120.35     121.00            0    0.00 %

    37       34.40      34.27      120.28     121.00            0    0.00 %



-----------------------------------------------------------------------------
Total Boardings in Minor Grid/Coastline Areas......             2   28.57 %
Total Boardings Outside Minor Grid Areas...........             5   71.43 %
Total Boardings....................................             7
```

Figure B.3 Minor Grid Area Boarding Frequency. (cont'd.)



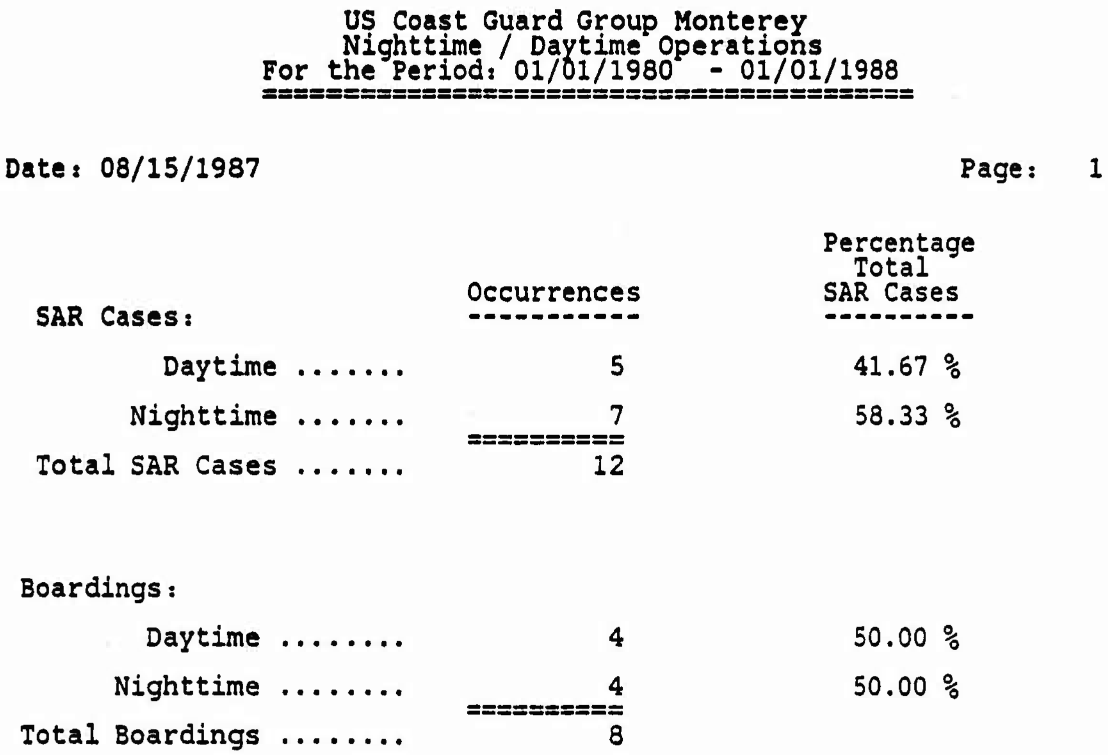
```
          US Coast Guard Group Monterey Boarding History
          ==============================================

                                                          Date-08/15/1987
                                                          Page-  1

   DATE        TIME     BONO     VESSEL HIN    BOLAT  BOLONG      POB
01/15/1980   5:00:00a  mry-001  111111111111   37.00N 121.52W      8
       V54 V55 V56 V57 V58 V59 V60 V61 V62 V63 V64 V65 V66 V67 V68
        x       x       x       x       x       x       x       x
    U69 U70 U71 U72 U73 U74 U75 U76 U77 U78  NOVIOLS  WEAPONS   ALCOHOL
                 x       x       x       x                 3        x
   owner/operator was abusive towards boarding officer

       BOARDING OFFICER - smith                     bm3
------------------------------------------------------------------------
   DATE        TIME     BONO     VESSEL HIN    BOLAT  BOLONG      POB
04/17/1985   2:30:00p  mry-008  111111111111   36.38N 121.52W      0
       V54 V55 V56 V57 V58 V59 V60 V61 V62 V63 V64 V65 V66 V67 V68
                x   x   x
    U69 U70 U71 U72 U73 U74 U75 U76 U77 U78  NOVIOLS  WEAPONS   ALCOHOL
             x                   x                         2        x
   owner/operator was uncooperative and abusive towards boarding officer

       BOARDING OFFICER - adams                     bm2
------------------------------------------------------------------------
```

Figure B.4 Vessel Boarding History.



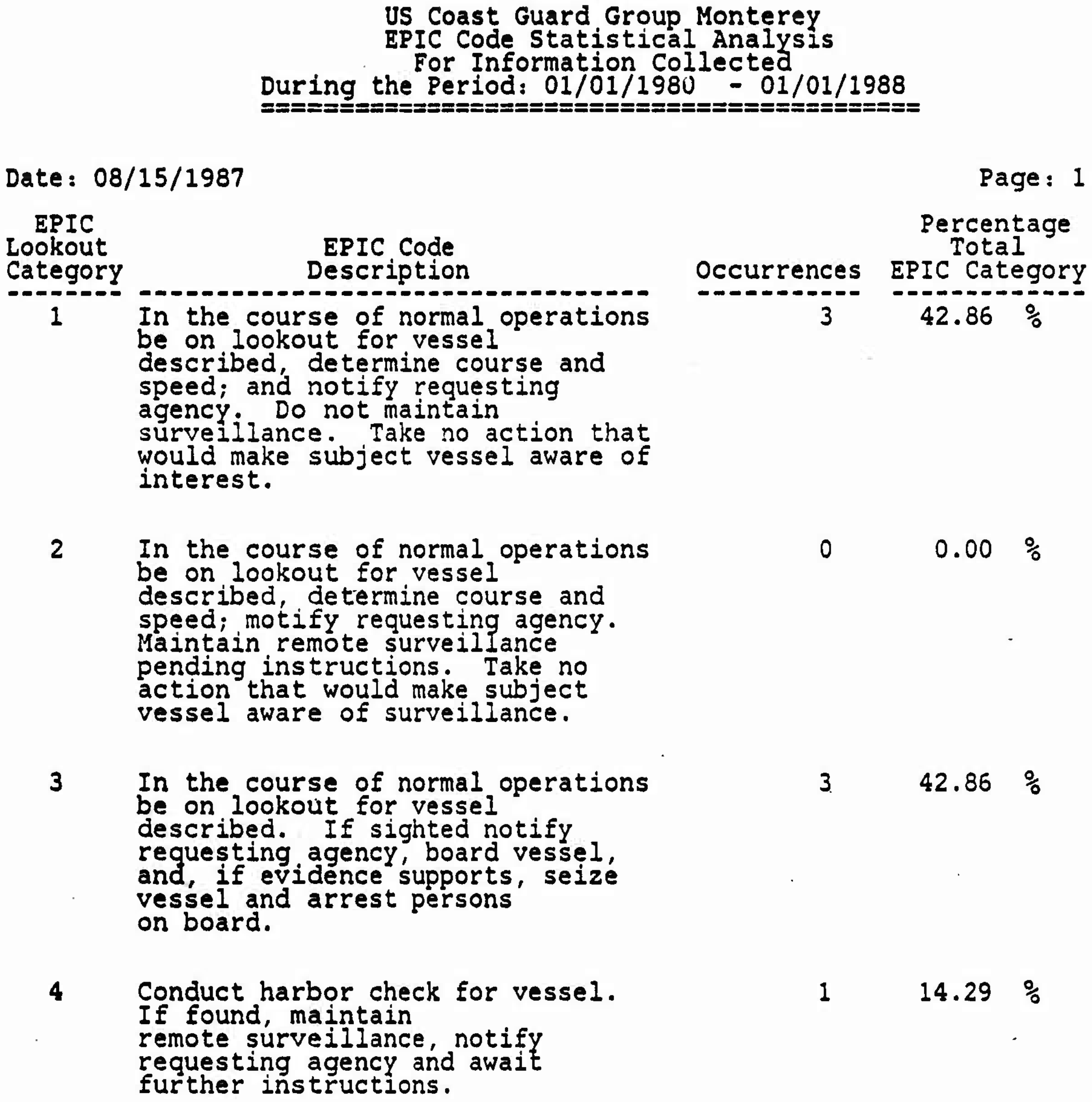
```
                 US Coast Guard Group Monterey
             Courtesy Motorboat Examinations (CME's)
          During the Period: 01/01/1980  - 01/01/1988
          ===========================================

Date: 08/15/1987                                                  Page: 1

                                          Percentage        Percentage
                                            Total             Total
CME Issue Year         Occurrences          CME's           Boardings
--------------         -----------        ----------        ----------
prior to 1970 ........           0          0.00 %            0.00 %
 1971 - 1975 .........           0          0.00 %            0.00 %
 1976 - 1979 .........           0          0.00 %            0.00 %
 1980 - 1981 .........           0          0.00 %            0.00 %
 1982 - 1983 .........           2         50.00 %           15.38 %
 1984 - 1985 ........            2         50.00 %           15.38 %
 1986 - 1987 .........           0          0.00 %            0.00 %
 1988 - 1989 .........           0          0.00 %            0.00 %
 after 1990 ..........           0          0.00 %            0.00 %
-------------------------------------------------------------------------
Total CME's ..........           4                           30.76 %
Total Boardings ......           8
```

Figure B.5   Courtesy Motorboat Examination Summary.



```
                   US Coast Guard Group Monterey
                   Nighttime / Daytime Operations
               For the Period: 01/01/1980  - 01/01/1988
               ========================================

Date: 08/15/1987                                            Page:   1

                                                  Percentage
                                                    Total
                                 Occurrences      SAR Cases
 SAR Cases:                      -----------      ----------

        Daytime .......               5             41.67 %

      Nighttime .......               7             58.33 %
                                 ==========
 Total SAR Cases .......             12



Boardings:

       Daytime ........               4             50.00 %

     Nighttime ........               4             50.00 %
                                 ==========
Total Boardings ........              8
```

Figure B.6 SAR and Boarding Day/Night Summary.



```
                         US Coast Guard Group Monterey
                         EPIC Code Statistical Analysis
                            For Information Collected
                    During the Period: 01/01/1980  - 01/01/1988
                    ===========================================

Date: 08/15/1987                                                  Page: 1

 EPIC                                                         Percentage
Lookout                 EPIC Code                                Total
Category               Description               Occurrences  EPIC Category
-------- ---------------------------------       -----------  -------------
   1     In the course of normal operations            3        42.86  %
         be on lookout for vessel
         described, determine course and
         speed; and notify requesting
         agency.  Do not maintain
         surveillance.  Take no action that
         would make subject vessel aware of
         interest.

   2     In the course of normal operations            0         0.00  %
         be on lookout for vessel
         described, determine course and
         speed; motify requesting agency.
         Maintain remote surveillance
         pending instructions.  Take no
         action that would make subject
         vessel aware of surveillance.

   3     In the course of normal operations            3        42.86  %
         be on lookout for vessel
         described.  If sighted notify
         requesting agency, board vessel,
         and, if evidence supports, seize
         vessel and arrest persons
         on board.

   4     Conduct harbor check for vessel.              1        14.29  %
         If found, maintain
         remote surveillance, notify
         requesting agency and await
         further instructions.
```

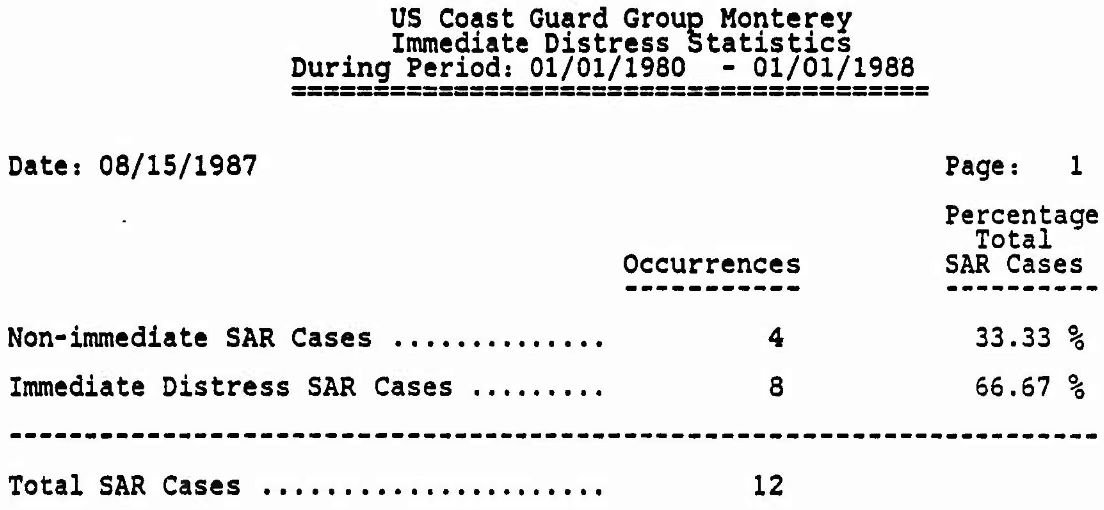
Figure B.7 EPIC Code Statistical Analysis.

```
Date: 08/15/1987                                                  Page: 2

                                                                 Percentage
  EPIC            EPIC Code                                         Total
Category         Description                      Occurrences EPIC Category
-------- ---------------------------------        ----------- ------------
   5     Conduct harbor check, board vessel             0         0.00  %
         if located and if
         evidence supports, seize vessel,
         arrest persons on board
         and report to requesting agency.


   6     Conduct search for vessel in most              0         0.00  %
         appropriate area based on
         information in paragraph 4 of
         lookout message.  If located,
         notify requesting agency,board
         vessel, and if evidence
         supports, seize vessel and arrest
         persons on board.

--------------------------------------------------------------------------

Total EPIC Occurrences .......................             7
```

Figure B.7 EPIC Code Statistical Analysis. (cont'd.)

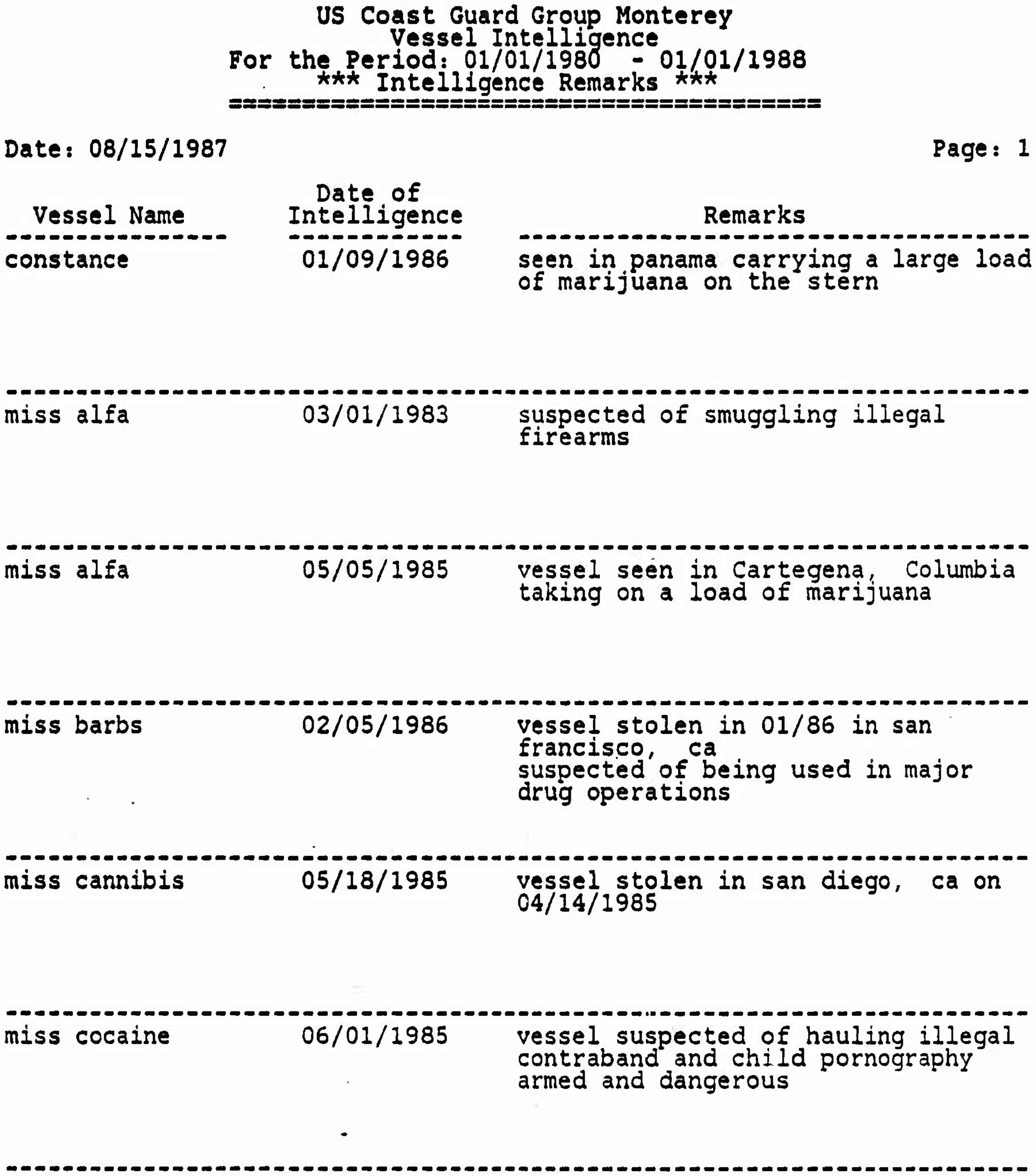
```
                US Coast Guard Group Monterey
                     Vessel Intelligence
           For the Period: 01/01/1980  - 01/01/1988
                   *** EPIC Code Summary ***
           ========================================

     Date: 08/15/1987                                         Page: 1

              Vessel                    Intelligence          EPIC
               Name                         Date              CODE
      -----------------               ------------          --------
      miss alfa                        03/01/1983              1

      miss alfa                        05/05/1985              1

      miss cocaine                     05/20/1986              1

      constance                        01/09/1986              3

      miss barbs                       02/05/1986              3

      miss cocaine                     06/01/1985              3

      miss cannibis                    05/18/1985              4
```

Figure B.8 Summary of EPIC Codes Assigned.



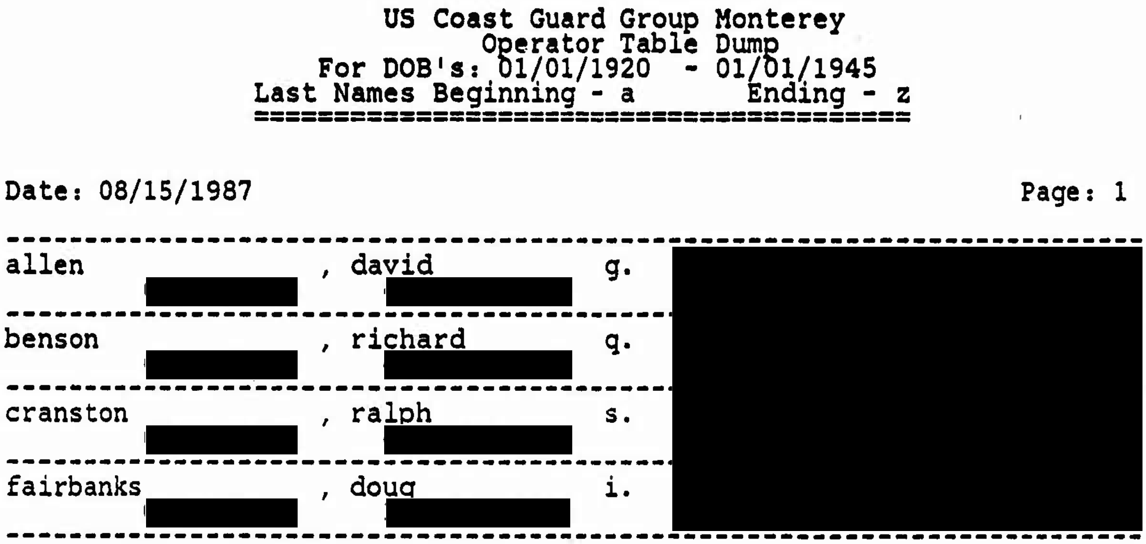
```
                    US Coast Guard Group Monterey
                    Immediate Distress Statistics
                 During Period: 01/01/1980  - 01/01/1988
                 ========================================

Date: 08/15/1987                                              Page:   1

                                                              Percentage
                                                                Total
                                            Occurrences       SAR Cases
                                            -----------       ----------

Non-immediate SAR Cases ..............           4              33.33 %

Immediate Distress SAR Cases .........           8              66.67 %

--------------------------------------------------------------------------

Total SAR Cases ......................          12
```

Figure B.9 Immediate Distress Statistics.



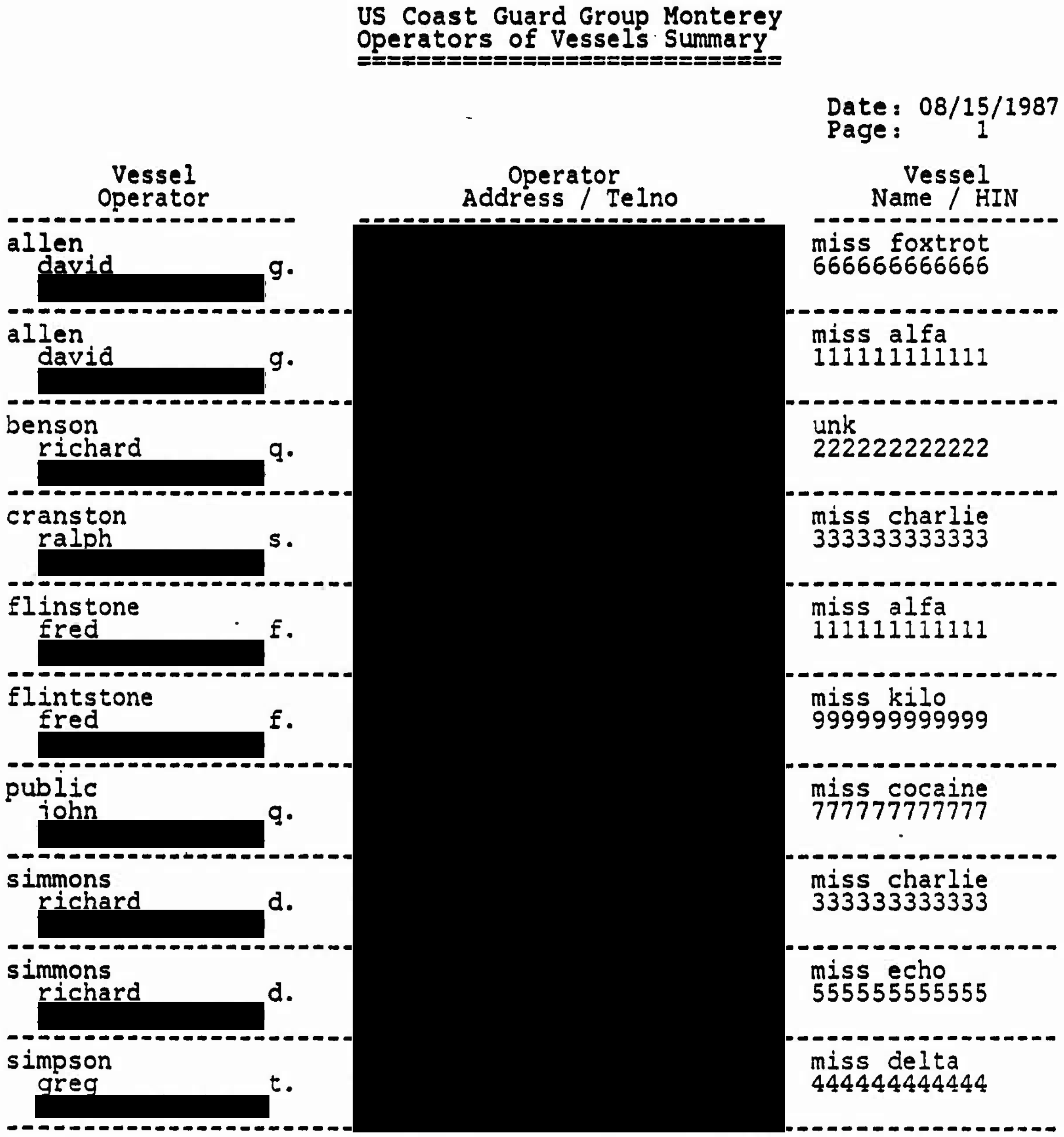
```
                   US Coast Guard Group Monterey
                        Vessel Intelligence
              For the Period: 01/01/1980  - 01/01/1988
                    *** Intelligence Remarks ***
              ==========================================

Date: 08/15/1987                                                  Page: 1

                      Date of
   Vessel Name      Intelligence                Remarks
 ---------------    ------------     ------------------------------------
constance            01/09/1986      seen in panama carrying a large load
                                     of marijuana on the stern


-------------------------------------------------------------------------
miss alfa            03/01/1983      suspected of smuggling illegal
                                     firearms


-------------------------------------------------------------------------
miss alfa            05/05/1985      vessel seen in Cartegena,  Columbia
                                     taking on a load of marijuana


-------------------------------------------------------------------------
miss barbs           02/05/1986      vessel stolen in 01/86 in san
                                     francisco,  ca
                                     suspected of being used in major
                                     drug operations

-------------------------------------------------------------------------
miss cannibis        05/18/1985      vessel stolen in san diego,  ca on
                                     04/14/1985


-------------------------------------------------------------------------
miss cocaine         06/01/1985      vessel suspected of hauling illegal
                                     contraband and child pornography
                                     armed and dangerous


-------------------------------------------------------------------------
```

Figure B.10 Vessel Intelligence Remarks Summary.



```
                    US Coast Guard Group Monterey
                         Operator Table Dump
                For DOB's: 01/01/1920  - 01/01/1945
            Last Names Beginning - a       Ending - z
            ==========================================

Date: 08/15/1987                                              Page: 1

-----------------------------------------------------------------------
allen      [redacted]  , david   [redacted]  g.   [redacted]
-----------------------------------------------------------------------
benson     [redacted]  , richard [redacted]  q.
-----------------------------------------------------------------------
cranston   [redacted]  , ralph   [redacted]  s.
-----------------------------------------------------------------------
fairbanks  [redacted]  , doug    [redacted]  i.
-----------------------------------------------------------------------
```

Figure B.11 Operator Listing.



```
                    US Coast Guard Group Monterey
                    Operators of Vessels Summary
                    ============================

                                                    Date: 08/15/1987
                                                    Page:      1

       Vessel                 Operator                 Vessel
      Operator           Address / Telno            Name / HIN
---------------------- ------------------------- ------------------
allen                  [redacted]                miss foxtrot
  david            g.                            666666666666
  [redacted]
---------------------                            ------------------
allen                                            miss alfa
  david            g.                            111111111111
  [redacted]
---------------------                            ------------------
benson                                           unk
  richard          q.                            222222222222
  [redacted]
---------------------                            ------------------
cranston                                         miss charlie
  ralph            s.                            333333333333
  [redacted]
---------------------                            ------------------
flinstone                                        miss alfa
  fred             f.                            111111111111
  [redacted]
---------------------                            ------------------
flintstone                                       miss kilo
  fred             f.                            999999999999
  [redacted]
---------------------                            ------------------
public                                           miss cocaine
  john             q.                            777777777777
  [redacted]
---------------------                            ------------------
simmons                                          miss charlie
  richard          d.                            333333333333
  [redacted]
---------------------                            ------------------
simmons                                          miss echo
  richard          d.                            555555555555
  [redacted]
---------------------                            ------------------
simpson                                          miss delta
  greg             t.                            444444444444
  [redacted]
---------------------                            ------------------
```

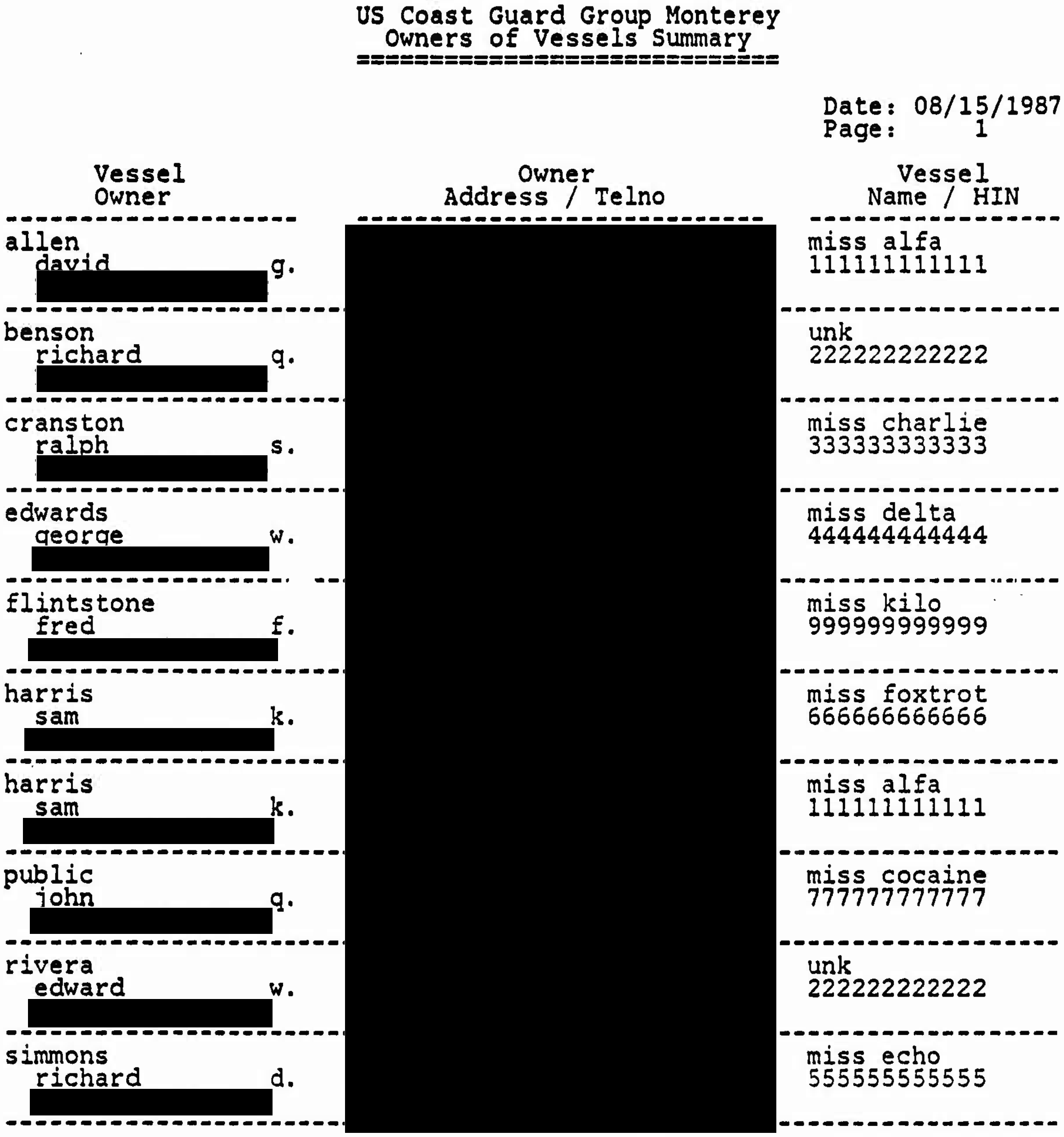
Figure B.12 Operators of Vessels Summary.

```
                    US Coast Guard Group Monterey
                    Operator Query Summary Report
                    =============================

                                                  Date: 01/01 1980

                     Last Name: flinstone
                    First Name: fred
                       Initial: f
                  [redacted]
```

Figure B.13 Operator Query Report.

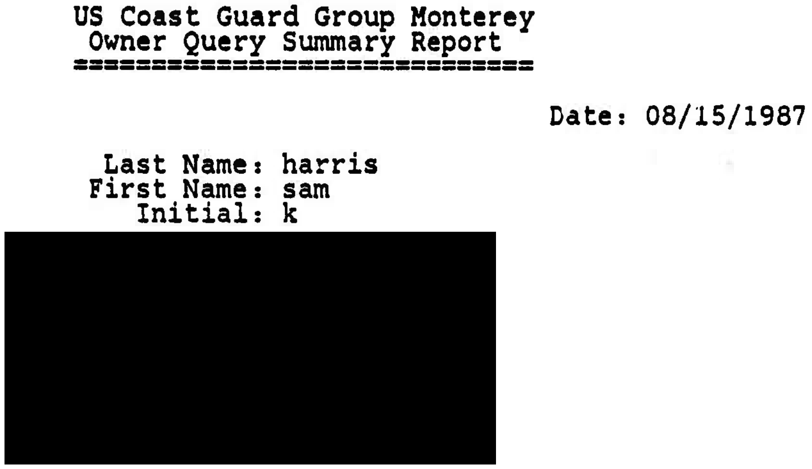
```
                    US Coast Guard Group Monterey
                          Owners Table Dump
                For DOB's: 01/01/1940  - 01/01/1960
               Last Names Beginning - a    Ending - z
               ======================================

Date: 08/15/1987                                                Page: 1

------------------------------------------------------------------------
allen      [redacted]  , david [redacted] g.  [redacted]
-------------------------------------------
edwards    [redacted]  , george [redacted] w.
-------------------------------------------
flintstone [redacted]  , fred  [redacted] f.
-------------------------------------------
harris     [redacted]  , sam   [redacted] k.
-------------------------------------------
williams   [redacted]  , scott [redacted] j.
-------------------------------------------
```

Figure B.14 Owners Listing.



```
                    US Coast Guard Group Monterey
                    Owners of Vessels Summary
                    =============================

                                                    Date: 08/15/1987
                                                    Page:      1

       Vessel                    Owner                    Vessel
       Owner               Address / Telno             Name / HIN
--------------------   -------------------------   -----------------
allen                  [redacted]                  miss alfa
  david           g.                               111111111111
  [redacted]
--------------------                               -----------------
benson                                             unk
  richard         q.                               222222222222
  [redacted]
--------------------                               -----------------
cranston                                           miss charlie
  ralph           s.                               333333333333
  [redacted]
--------------------                               -----------------
edwards                                            miss delta
  george          w.                               444444444444
  [redacted]
--------------------                               -----------------
flintstone                                         miss kilo
  fred            f.                               999999999999
  [redacted]
--------------------                               -----------------
harris                                             miss foxtrot
  sam             k.                               666666666666
  [redacted]
--------------------                               -----------------
harris                                             miss alfa
  sam             k.                               111111111111
  [redacted]
--------------------                               -----------------
public                                             miss cocaine
  john            q.                               777777777777
  [redacted]
--------------------                               -----------------
rivera                                             unk
  edward          w.                               222222222222
  [redacted]
--------------------                               -----------------
simmons                                            miss echo
  richard         d.                               555555555555
  [redacted]
--------------------                               -----------------
```

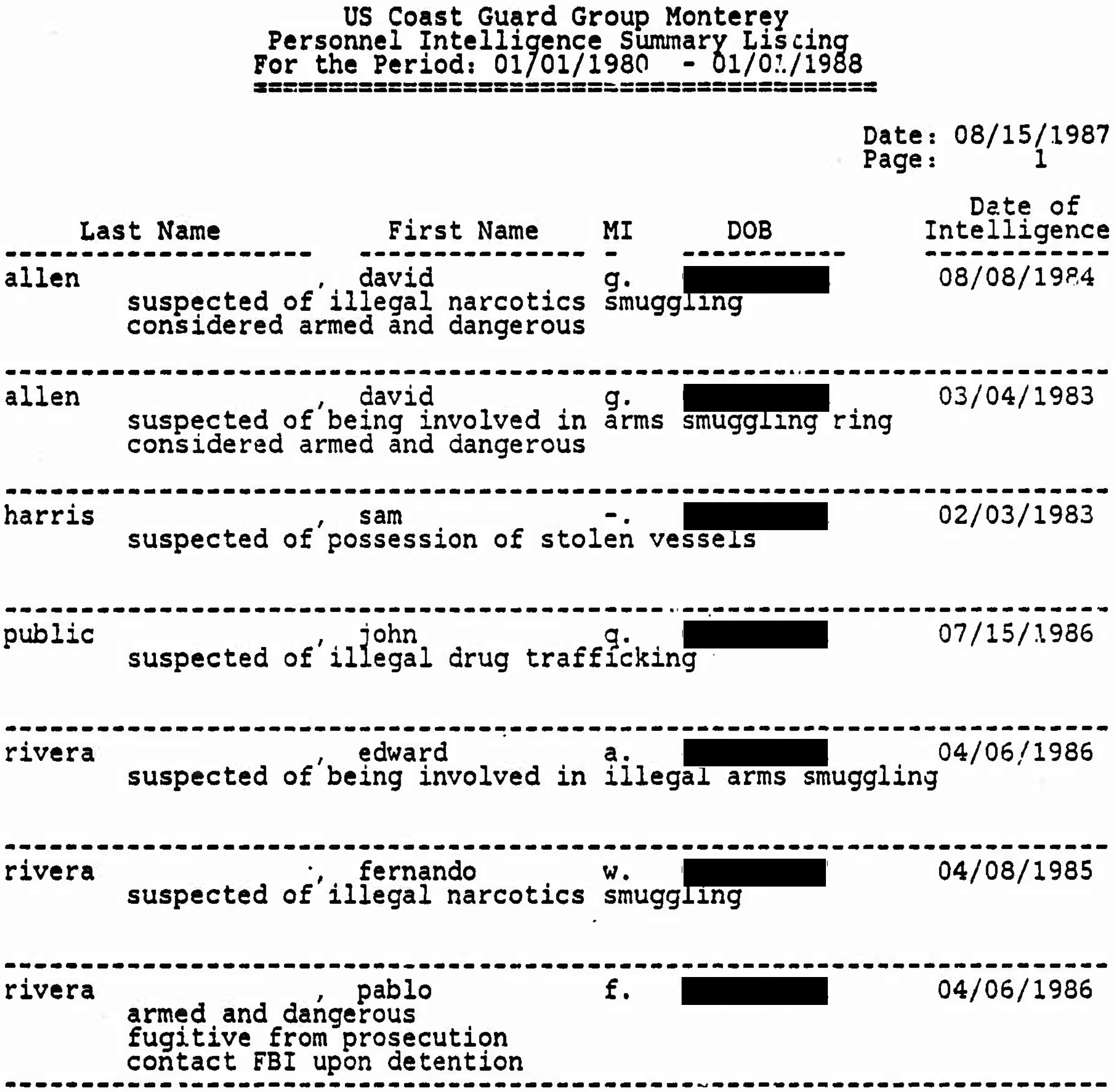
Figure B.15 Owners of Vessels Summary.



```
              US Coast Guard Group Monterey
               Owner Query Summary Report
              ============================

                                                 Date: 08/15/1987

                Last Name: harris
               First Name: sam
                  Initial: k
```

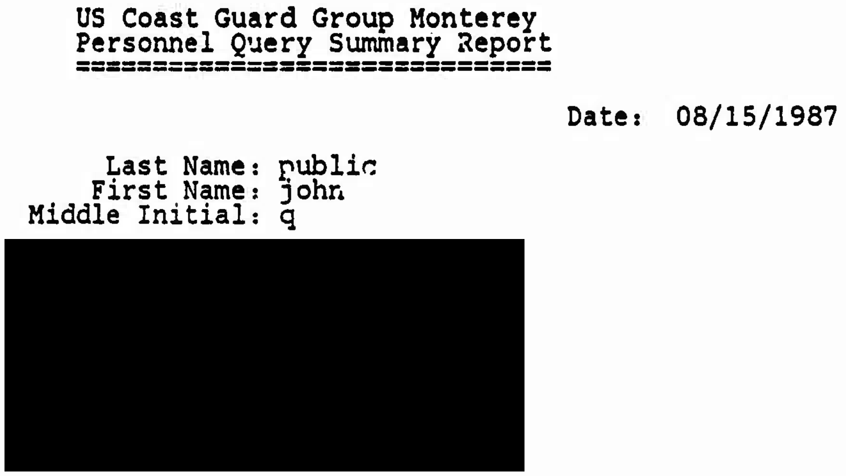
[redacted]

Figure B.16 Owner Query Report.



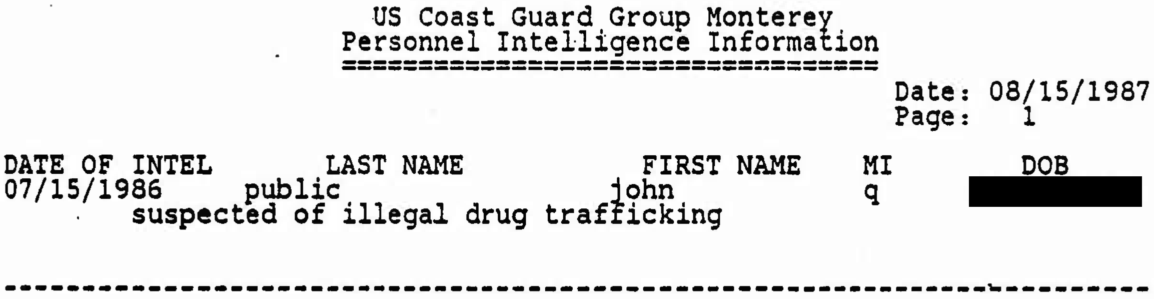
```
                    US Coast Guard Group Monterey
               Personnel Intelligence Summary Liscing
              For the Period: 01/01/1980  - 01/01/1988
              ========================================

                                                          Date: 08/15/1987
                                                          Page:        1
                                                                Date of
    Last Name             First Name       MI      DOB        Intelligence
-------------------- ---------------- -  -----------      ------------
allen                 , david          g.  [redacted]        08/08/1984
        suspected of illegal narcotics smuggling
        considered armed and dangerous

------------------------------------------------------------------------
allen                 , david          g.  [redacted]        03/04/1983
        suspected of being involved in arms smuggling ring
        considered armed and dangerous

------------------------------------------------------------------------
harris                , sam            -.  [redacted]        02/03/1983
        suspected of possession of stolen vessels

------------------------------------------------------------------------
public                , john           q.  [redacted]        07/15/1986
        suspected of illegal drug trafficking

------------------------------------------------------------------------
rivera                , edward         a.  [redacted]        04/06/1986
        suspected of being involved in illegal arms smuggling

------------------------------------------------------------------------
rivera                , fernando       w.  [redacted]        04/08/1985
        suspected of illegal narcotics smuggling

------------------------------------------------------------------------
rivera                , pablo          f.  [redacted]        04/06/1986
        armed and dangerous
        fugitive from prosecution
        contact FBI upon detention
------------------------------------------------------------------------
```

Figure B.17 Personnel Intelligence Listing.



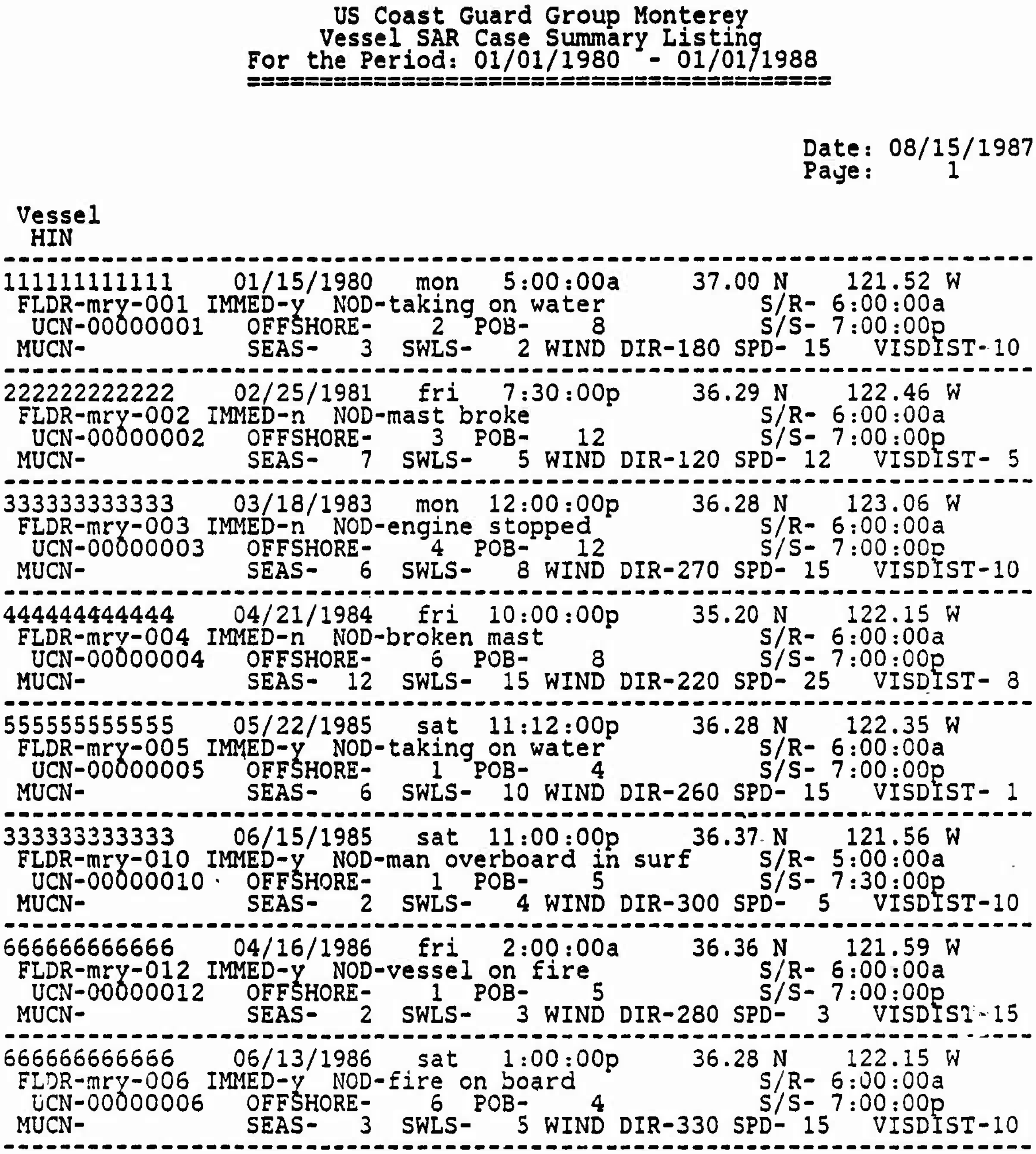
```
          US Coast Guard Group Monterey
          Personnel Query Summary Report
          ==============================

                                      Date:  08/15/1987

      Last Name: public
     First Name: john
 Middle Initial: q
[redacted]
```

Figure B.18 Personnel Query Report.



```
                     US Coast Guard Group Monterey
                   Personnel Intelligence Information
                   ==================================
                                                  Date: 08/15/1987
                                                  Page:    1

DATE OF INTEL        LAST NAME           FIRST NAME    MI        DOB
07/15/1986     public                   john           q     [redacted]
        suspected of illegal drug trafficking

----------------------------------------------------------------------
```

Figure B.19 Personnel Intelligence Report.

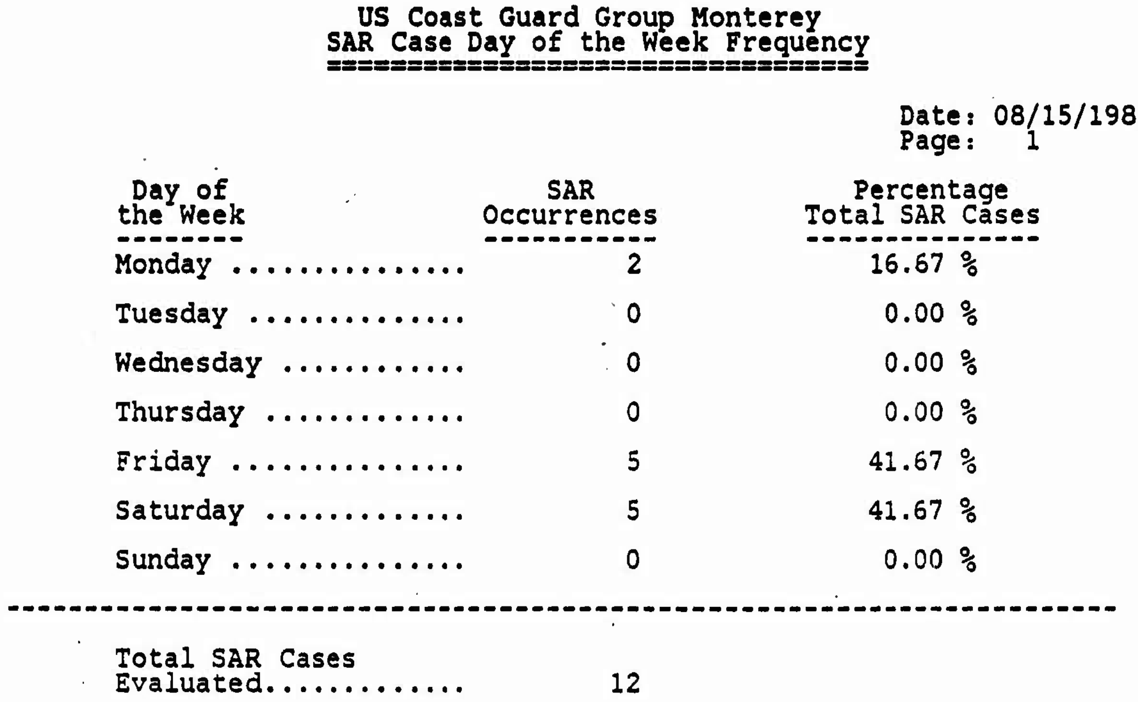


```
                    US Coast Guard Group Monterey
                    Vessel SAR Case Summary Listing
                For the Period: 01/01/1980  - 01/01/1988
                ========================================

                                                       Date: 08/15/1987
                                                       Page:      1
  Vessel
   HIN
------------------------------------------------------------------------
111111111111    01/15/1980   mon    5:00:00a     37.00 N     121.52 W
 FLDR-mry-001 IMMED-y  NOD-taking on water             S/R- 6:00:00a
  UCN-00000001   OFFSHORE-    2  POB-    8             S/S- 7:00:00p
 MUCN-           SEAS-   3  SWLS-   2 WIND DIR-180 SPD- 15   VISDIST-10
------------------------------------------------------------------------
222222222222    02/25/1981   fri    7:30:00p     36.29 N     122.46 W
 FLDR-mry-002 IMMED-n  NOD-mast broke                  S/R- 6:00:00a
  UCN-00000002   OFFSHORE-    3  POB-   12             S/S- 7:00:00p
 MUCN-           SEAS-   7  SWLS-   5 WIND DIR-120 SPD- 12   VISDIST- 5
------------------------------------------------------------------------
333333333333    03/18/1983   mon   12:00:00p     36.28 N     123.06 W
 FLDR-mry-003 IMMED-n  NOD-engine stopped              S/R- 6:00:00a
  UCN-00000003   OFFSHORE-    4  POB-   12             S/S- 7:00:00p
 MUCN-           SEAS-   6  SWLS-   8 WIND DIR-270 SPD- 15   VISDIST-10
------------------------------------------------------------------------
444444444444    04/21/1984   fri   10:00:00p     35.20 N     122.15 W
 FLDR-mry-004 IMMED-n  NOD-broken mast                 S/R- 6:00:00a
  UCN-00000004   OFFSHORE-    6  POB-    8             S/S- 7:00:00p
 MUCN-           SEAS-  12  SWLS-  15 WIND DIR-220 SPD- 25   VISDIST- 8
------------------------------------------------------------------------
555555555555    05/22/1985   sat   11:12:00p     36.28 N     122.35 W
 FLDR-mry-005 IMMED-y  NOD-taking on water             S/R- 6:00:00a
  UCN-00000005   OFFSHORE-    1  POB-    4             S/S- 7:00:00p
 MUCN-           SEAS-   6  SWLS-  10 WIND DIR-260 SPD- 15   VISDIST- 1
------------------------------------------------------------------------
333333333333    06/15/1985   sat   11:00:00p     36.37 N     121.56 W
 FLDR-mry-010 IMMED-y  NOD-man overboard in surf       S/R- 5:00:00a
  UCN-00000010   OFFSHORE-    1  POB-    5             S/S- 7:30:00p
 MUCN-           SEAS-   2  SWLS-   4 WIND DIR-300 SPD-  5   VISDIST-10
------------------------------------------------------------------------
666666666666    04/16/1986   fri    2:00:00a     36.36 N     121.59 W
 FLDR-mry-012 IMMED-y  NOD-vessel on fire              S/R- 6:00:00a
  UCN-00000012   OFFSHORE-    1  POB-    5             S/S- 7:00:00p
 MUCN-           SEAS-   2  SWLS-   3 WIND DIR-280 SPD-  3   VISDIST-15
------------------------------------------------------------------------
666666666666    06/13/1986   sat    1:00:00p     36.28 N     122.15 W
 FLDR-mry-006 IMMED-y  NOD-fire on board               S/R- 6:00:00a
  UCN-00000006   OFFSHORE-    6  POB-    4             S/S- 7:00:00p
 MUCN-           SEAS-   3  SWLS-   5 WIND DIR-330 SPD- 15   VISDIST-10
------------------------------------------------------------------------
```

Figure B.20 SAR Listing.

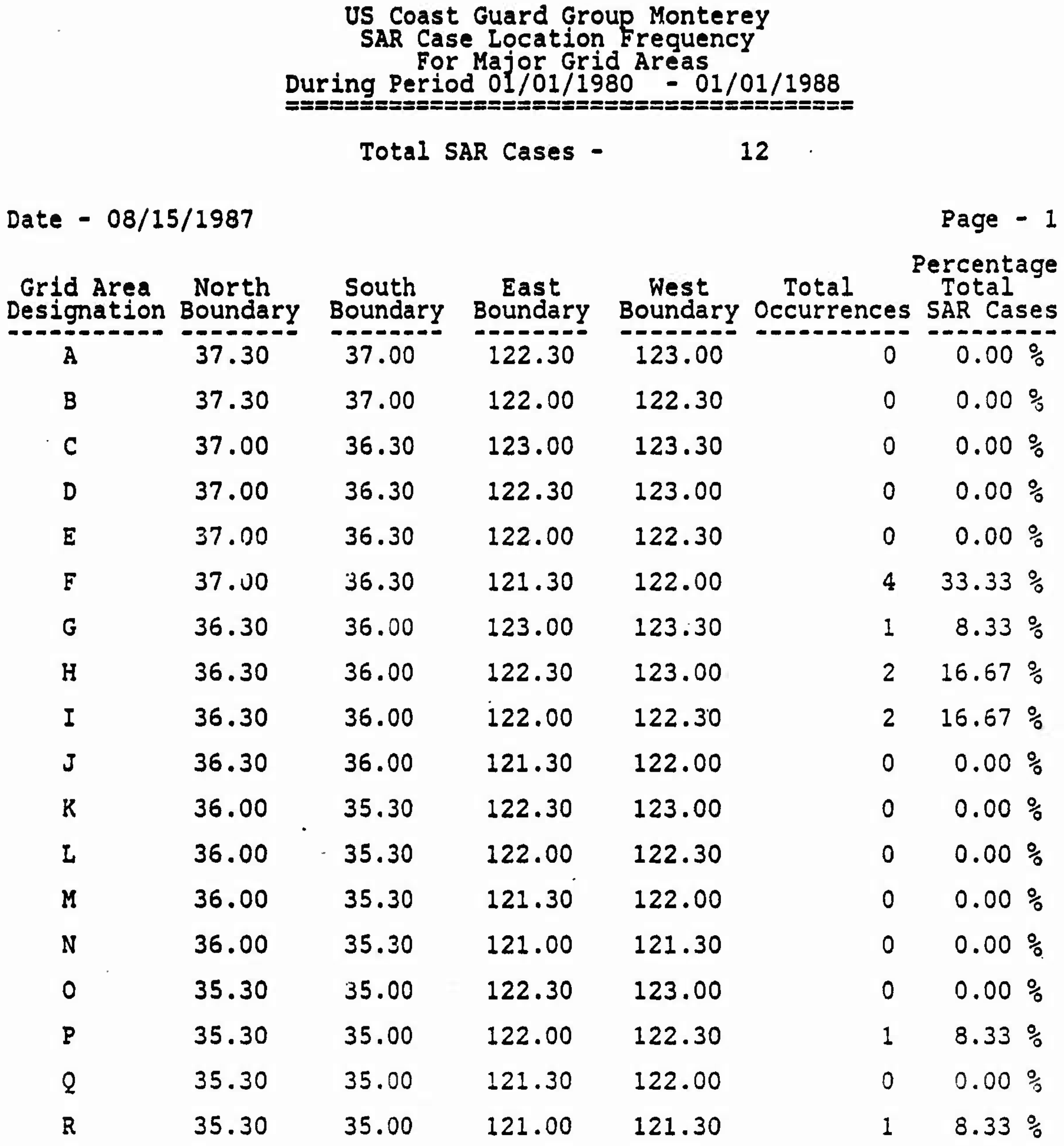
```
                      US Coast Guard Group Monterey
                      SAR Case Frequency Statistics
                         For Months and Years
                        01/01/1980  - 12/30/1986
                      ============================

                                                      Date: 08/15/1987
                                                      Page:     1

                                                    Percentage
                                  Total Number        Total
      Month                        SAR Cases        SAR Cases
     -----------                  ------------      ----------
      January ..............           1               9.09 %
      February .............           1               9.09 %
      March ................           1               9.09 %
      April ................           2              18.18 %
       May .................           1               9.09 %
      June .................           3              27.27 %
      July .................           1               9.09 %
     August ................           1               9.09 %
     September .............           0               0.00 %
      October ..............           0               0.00 %
     November ..............           0               0.00 %
     December ..............           0               0.00 %

---------------------------------------------------------------------

                                           Total Number
                   Year                     SAR Cases
                   ----                    ------------
                   1980 .................        1

                   1981 .................        1

                   1982 .................        0

                   1983 .................        1

                   1984 .................        2

                   1985 .................        2

                   1986 .................        4

---------------------------------------------------------------------

              Total SAR Cases ...........       11
```

Figure B.21 SAR Months and Years Frequency Statistics.



```
                 US Coast Guard Group Monterey
               SAR Case Day of the Week Frequency
               ==================================

                                                   Date: 08/15/1987
                                                   Page:    1

  Day of                     SAR               Percentage
 the Week                Occurrences        Total SAR Cases
 --------                -----------        ---------------
 Monday ...............        2                16.67 %

 Tuesday ..............        0                 0.00 %

 Wednesday ............        0                 0.00 %

 Thursday .............        0                 0.00 %

 Friday ...............        5                41.67 %

 Saturday .............        5                41.67 %

 Sunday ...............        0                 0.00 %

--------------------------------------------------------------------

 Total SAR Cases
 Evaluated.............       12
```

Figure B.22 SAR Day of Week Frequency Statistics.



```
                    US Coast Guard Group Monterey
                    SAR Case Location Frequency
                       For Major Grid Areas
                During Period 01/01/1980  - 01/01/1988
                =======================================

                      Total SAR Cases -         12

Date - 08/15/1987                                                 Page - 1

                                                                Percentage
 Grid Area   North     South     East      West      Total       Total
Designation Boundary  Boundary  Boundary  Boundary Occurrences SAR Cases
----------- --------  --------  --------  -------- ----------- ---------
     A        37.30     37.00    122.30    123.00            0    0.00 %

     B        37.30     37.00    122.00    122.30            0    0.00 %

     C        37.00     36.30    123.00    123.30            0    0.00 %

     D        37.00     36.30    122.30    123.00            0    0.00 %

     E        37.00     36.30    122.00    122.30            0    0.00 %

     F        37.00     36.30    121.30    122.00            4   33.33 %

     G        36.30     36.00    123.00    123.30            1    8.33 %

     H        36.30     36.00    122.30    123.00            2   16.67 %

     I        36.30     36.00    122.00    122.30            2   16.67 %

     J        36.30     36.00    121.30    122.00            0    0.00 %

     K        36.00     35.30    122.30    123.00            0    0.00 %

     L        36.00     35.30    122.00    122.30            0    0.00 %

     M        36.00     35.30    121.30    122.00            0    0.00 %

     N        36.00     35.30    121.00    121.30            0    0.00 %

     O        35.30     35.00    122.30    123.00            0    0.00 %

     P        35.30     35.00    122.00    122.30            1    8.33 %

     Q        35.30     35.00    121.30    122.00            0    0.00 %

     R        35.30     35.00    121.00    121.30            1    8.33 %
```

Figure B.23 Major Grid Area SAR Frequency.

```
Date - 08/15/1987                                                       Page - 2

                                                                      Percentage
 Grid Area   North      South       East       West       Total       Total
Designation Boundary   Boundary   Boundary   Boundary  Occurrences  SAR Cases
----------- --------   --------   --------   --------  -----------  ---------
    S        35.30      35.00      120.30     121.00            0    0.00 %

    T        35.00      34.30      122.00     122.30            0    0.00 %

    U        35.00      34.30      121.30     122.00            0    0.00 %

    V        35.00      34.30      121.00     121.30            0    0.00 %

    W        35.00      34.30      120.30     121.00            0    0.00 %

    X        34.30      34.00      121.30     122.00            0    0.00 %

    Y        34.30      34.00      121.00     121.30            0    0.00 %



-----------------------------------------------------------------------
Total SAR Cases in Major Grid Areas................            11   91.67 %
Total SAR Cases Outside Grid Areas.................             1    8.33 %
Total SAR Cases....................................            12
```

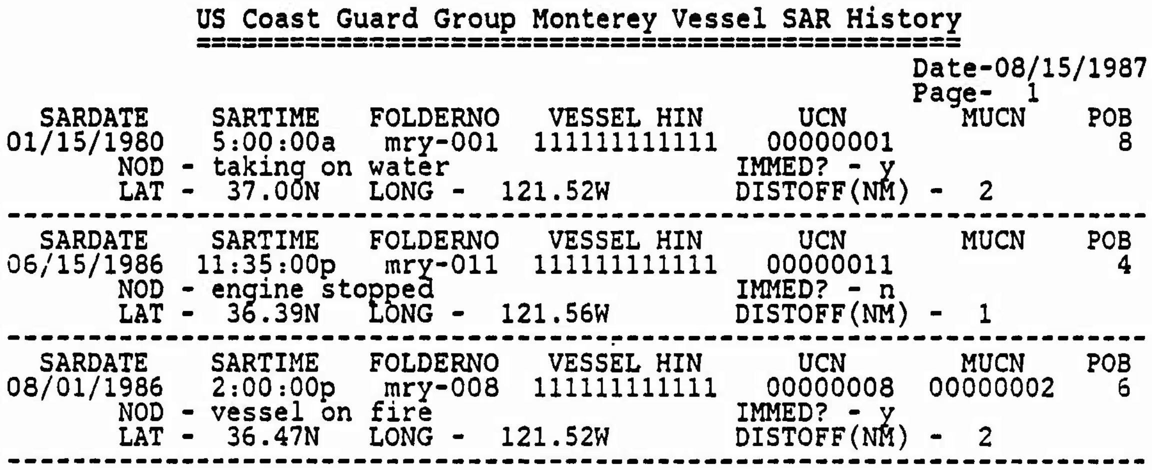
Figure B.23 Major Grid Area SAR Frequency. (cont'd.)

```
                       US Coast Guard Group Monterey
                        SAR Case Location Frequency
                      For Minor Grid Areas (Coastline)
                   During Period 01/01/1980  - 01/01/1988
                   ======================================

                        Total SAR Cases -        12

Date - 08/15/1987                                                      Page - 1
                                                                     Percentage
 Grid Area   North      South      East      West      Total           Total
Designation Boundary  Boundary  Boundary  Boundary Occurrences  SAR Cases
----------- --------  --------  --------  -------- -----------  ---------
    01        37.10     37.05    122.20    122.25            0     0.00 %

    02        37.10     37.05    122.15    122.20            0     0.00 %

    03        37.05     37.00    122.20    122.25            0     0.00 %

    04        37.05     37.00    122.15    122.20            0     0.00 %

    05        37.05     37.00    122.10    122.15            0     0.00 %

    06        37.00     36.55    122.15    122.20            0     0.00 %

    07        37.00     36.55    122.10    122.15            0     0.00 %

    08        37.00     36.55    122.05    122.10            0     0.00 %

    09        37.00     36.55    122.00    122.05            0     0.00 %

    10        37.00     36.55    121.55    122.00            0     0.00 %

    11        37.00     36.55    121.50    121.55            0     0.00 %

    12        36.55     36.50    122.00    122.05            0     0.00 %

    13        36.55     36.50    121.55    122.00            0     0.00 %

    14        36.55     36.50    121.50    121.55            0     0.00 %

    15        36.55     36.50    121.45    121.50            0     0.00 %

    16        36.50     36.45    122.00    122.05            0     0.00 %

    17        36.50     36.45    121.55    122.00            0     0.00 %

    18        36.50     36.45    121.50    121.55            1     8.33 %

    19        36.50     36.45    121.45    121.50            0     0.00 %
```

Figure B.24 Minor Grid Area SAR Frequency.

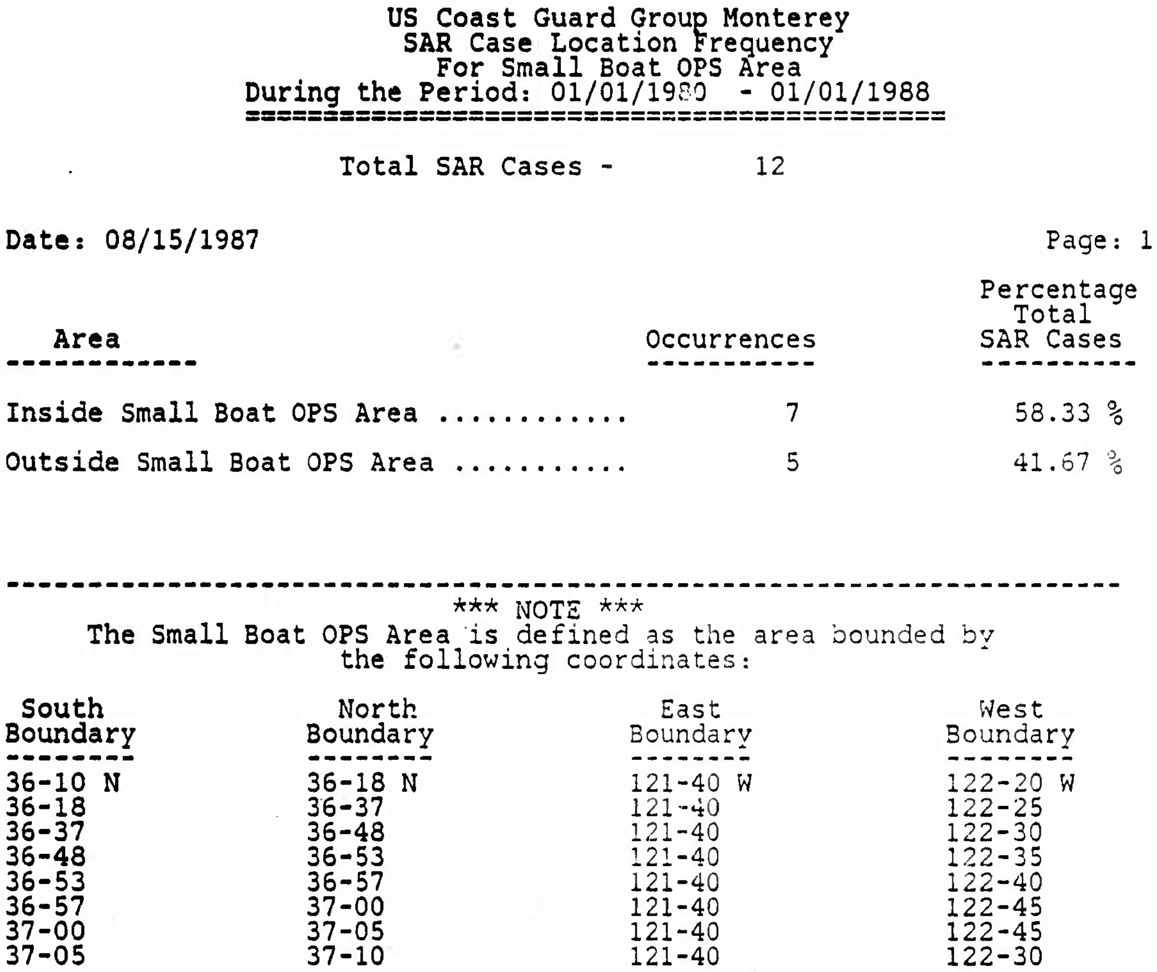
```
Date - 08/15/1987                                                    Page - 2

                                                                 Percentage
 Grid Area   North      South      East       West      Total       Total
Designation Boundary   Boundary   Boundary   Boundary Occurrences SAR Cases
----------- --------   --------   --------   -------- ----------- ---------
    20       36.45      36.40     122.00     122.05           0     0.00 %
    21       36.45      36.40     121.55     122.00           0     0.00 %
    22       36.45      36.40     121.50     121.55           0     0.00 %
    23       36.45      36.40     121.45     121.50           0     0.00 %
    24       36.40      36.35     122.00     122.05           0     0.00 %
    25       36.40      36.35     121.55     122.00           3    25.00 %
    26       36.40      36.35     121.50     121.55           0     0.00 %
    27       36.40      36.35     121.45     121.50           0     0.00 %
    28       36.35      36.30     121.55     122.05           0     0.00 %
    29       36.30      36.25     121.53     122.05           0     0.00 %
    30       36.25      36.20     121.52     122.05           0     0.00 %
    31       36.20      36.15     121.50     122.05           0     0.00 %
    32       36.15      36.00     121.30     122.00           0     0.00 %
    33       36.00      35.40     121.15     121.45           0     0.00 %
    34       35.40      35.20     120.45     121.30           1     8.33 %
    35       35.20      35.00     120.35     121.10           0     0.00 %
    36       35.00      34.40     120.35     121.00           0     0.00 %
    37       34.40      34.27     120.28     121.00           0     0.00 %


---------------------------------------------------------------------------
Total SAR Cases in Minor Grid/Coastline Areas......           5    41.67 %
Total SAR Cases Outside Grid Areas.................           7    58.33 %
Total SAR Cases....................................          12
```

Figure B.24 Minor Grid Area SAR Frequency. (cont'd.)



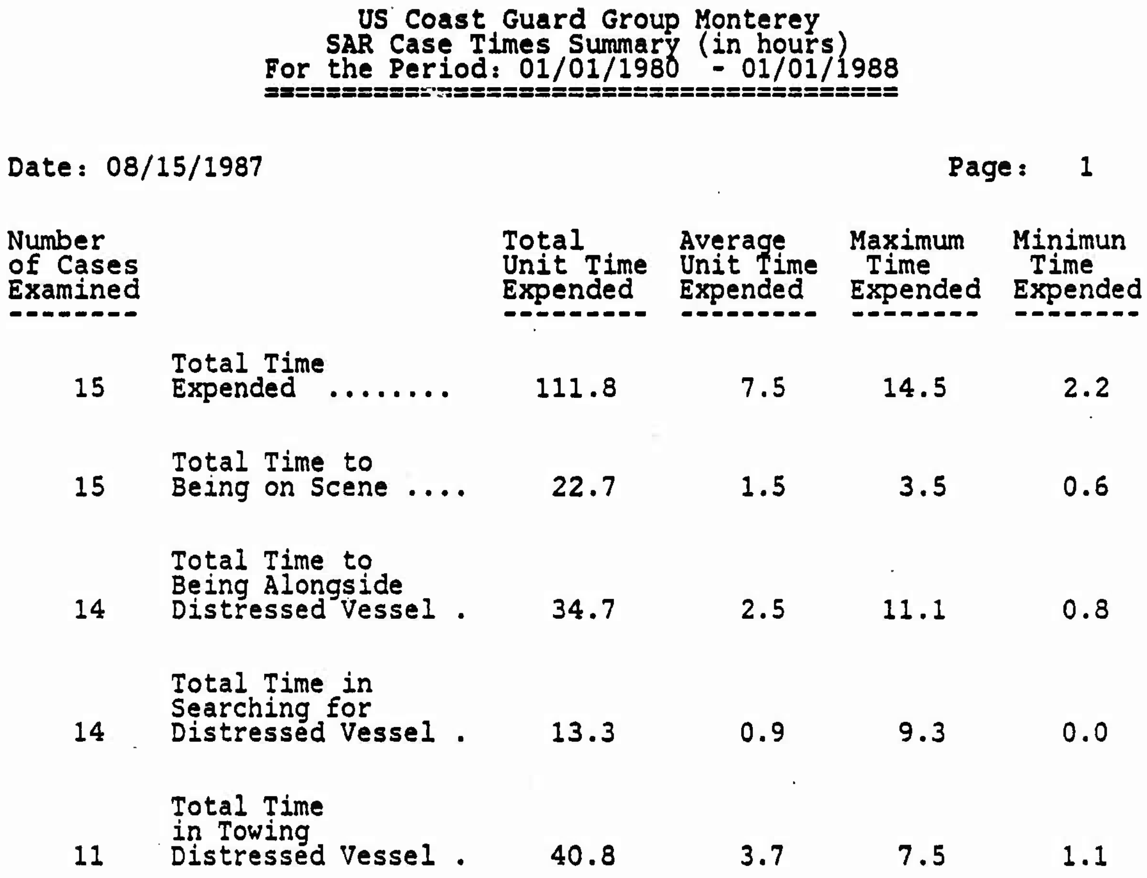
```
                     US Coast Guard Group Monterey Vessel SAR History
                     ===============================================
                                                                      Date-08/15/1987
                                                                      Page-  1
  SARDATE     SARTIME    FOLDERNO    VESSEL HIN       UCN         MUCN      POB
01/15/1980   5:00:00a    mry-001   111111111111   00000001                    8
      NOD - taking on water                     IMMED? - y
      LAT -  37.00N    LONG -  121.52W          DISTOFF(NM) -  2
-------------------------------------------------------------------------------
  SARDATE     SARTIME    FOLDERNO    VESSEL HIN       UCN         MUCN      POB
06/15/1986  11:35:00p    mry-011   111111111111   00000011                    4
      NOD - engine stopped                      IMMED? - n
      LAT -  36.39N    LONG -  121.56W          DISTOFF(NM) -  1
-------------------------------------------------------------------------------
  SARDATE     SARTIME    FOLDERNO    VESSEL HIN       UCN         MUCN      POB
08/01/1986   2:00:00p    mry-008   111111111111   00000008   00000002        6
      NOD - vessel on fire                      IMMED? - y
      LAT -  36.47N    LONG -  121.52W          DISTOFF(NM) -  2
-------------------------------------------------------------------------------
```

Figure B.46 Vessel Query SAR History.



```
                    US Coast Guard Group Monterey
                    SAR Case Location Frequency
                       For Small Boat OPS Area
              During the Period: 01/01/1980  - 01/01/1988
              ============================================

                     Total SAR Cases -        12

Date: 08/15/1987                                           Page: 1

                                                        Percentage
                                                          Total
   Area                                    Occurrences   SAR Cases
------------                               -----------   ----------

Inside Small Boat OPS Area ............          7          58.33 %

Outside Small Boat OPS Area ...........          5          41.67 %



----------------------------------------------------------------------
                          *** NOTE ***
     The Small Boat OPS Area is defined as the area bounded by
                    the following coordinates:

 South               North                East                West
Boundary            Boundary             Boundary            Boundary
--------            --------             --------            --------
36-10 N             36-18 N              121-40 W            122-20 W
36-18               36-37                121-40              122-25
36-37               36-48                121-40              122-30
36-48               36-53                121-40              122-35
36-53               36-57                121-40              122-40
36-57               37-00                121-40              122-45
37-00               37-05                121-40              122-45
37-05               37-10                121-40              122-30
```

Figure B.26 Small Boat Operations Area SAR Frequency.

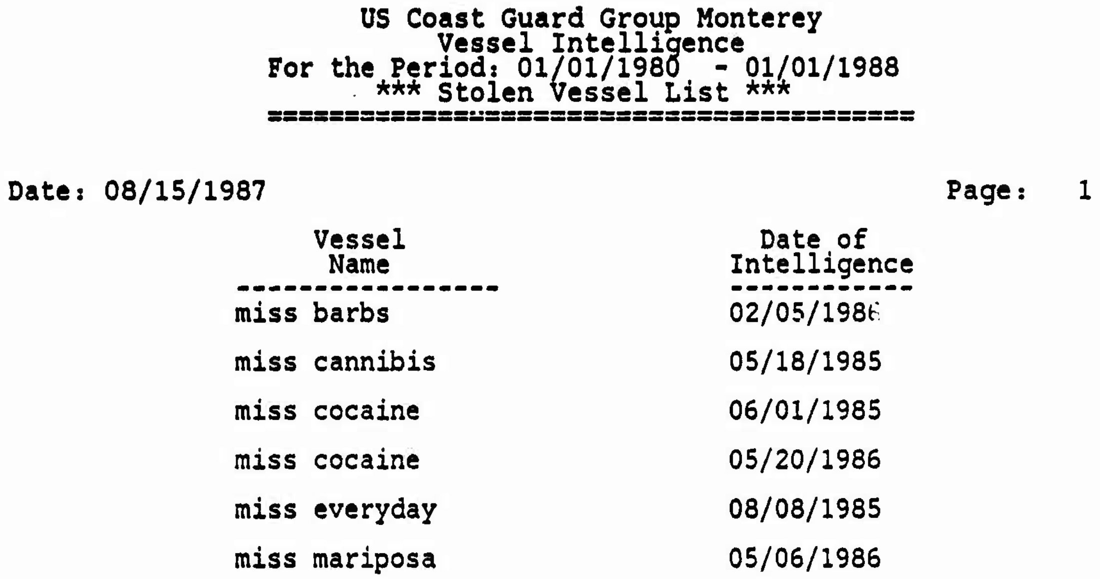


```
                   US Coast Guard Group Monterey
                SAR Case Times Summary (in hours)
            For the Period: 01/01/1980  - 01/01/1988
            ========================================

Date: 08/15/1987                                          Page:    1

Number                            Total      Average    Maximum   Minimun
of Cases                          Unit Time  Unit Time   Time      Time
Examined                          Expended   Expended   Expended  Expended
--------                          ---------  ---------  --------  --------

    15    Total Time
          Expended ........        111.8        7.5       14.5       2.2

    15    Total Time to
          Being on Scene ....       22.7        1.5        3.5       0.6

    14    Total Time to
          Being Alongside
          Distressed Vessel .       34.7        2.5       11.1       0.8

    14    Total Time in
          Searching for
          Distressed Vessel .       13.3        0.9        9.3       0.0

    11    Total Time
          in Towing
          Distressed Vessel .       40.8        3.7        7.5       1.1
```

Figure B.27 SAR Unit Time Summary.

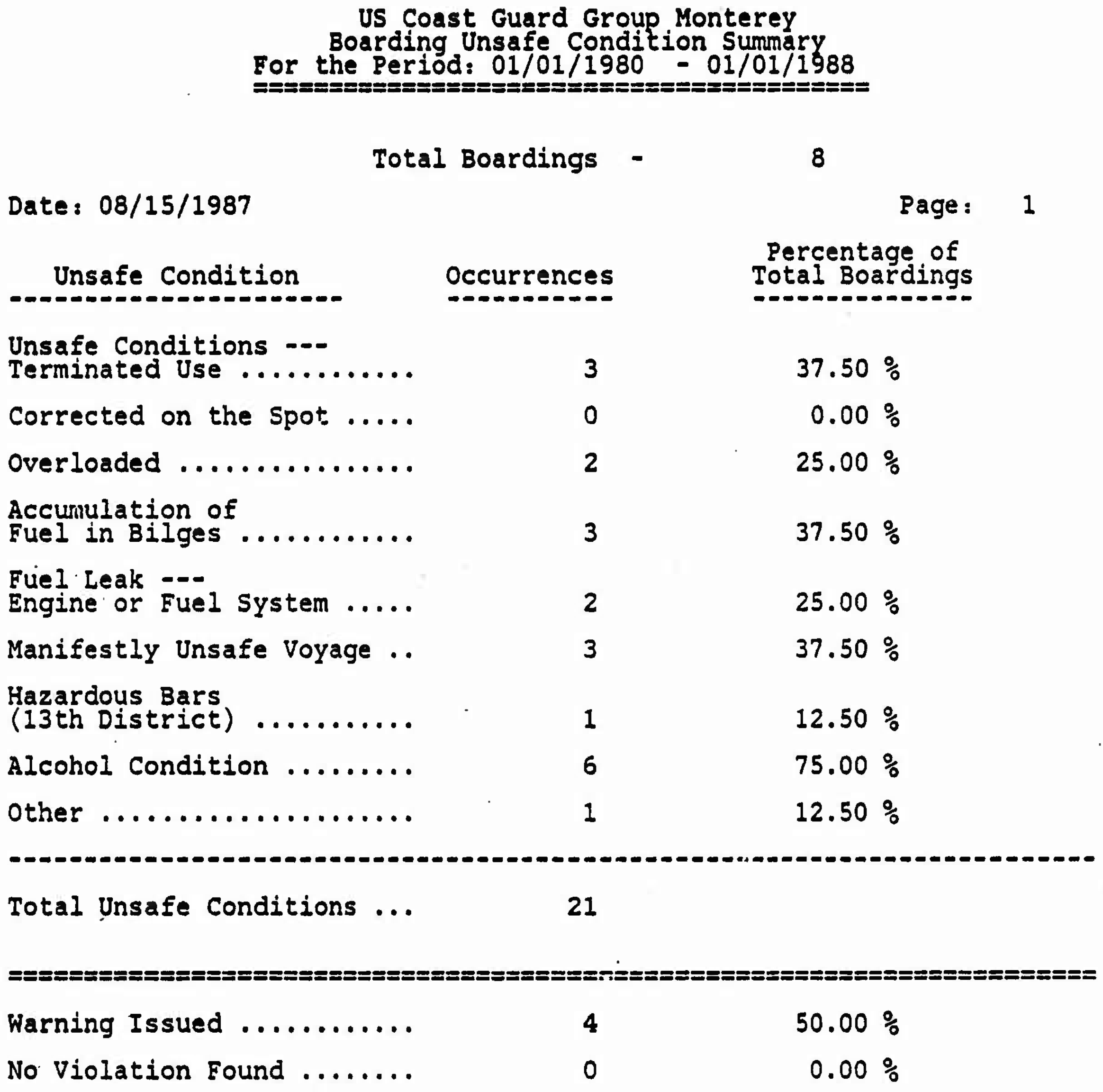
```
                    US Coast Guard Group Monterey
                     Boarding Location Frequency
                       For Small Boat OPS Area
                 During the Period: 01/01/1980  - 01/01/1988
                 ==========================================

                    Total Boardings -         8

Date: 08/15/1987                                         Page: 1

                                                       Percentage
                                                         Total
   Area                                 Occurrences    Boardings
------------                            -----------    ----------

Inside Small Boat OPS Area ............           3       37.50 %

Outside Small Boat OPS Area ...........           5       62.50 %


--------------------------------------------------------------------
                            *** NOTE ***
     The Small Boat OPS Area is defined as the area bounded by
                      the following coordinates:

 South                North                East                West
Boundary             Boundary             Boundary            Boundary
--------             --------             --------            --------
36-10 N              36-18 N              121-40 W            122-20 W
36-18                36-37                121-40              122-25
36-37                36-48                121-40              122-30
36-48                36-53                121-40              122-35
36-53                36-57                121-40              122-40
36-57                37-00                121-40              122-45
37-00                37-05                121-40              122-45
37-05                37-10                121-40              122-30
```

Figure B.47 Small Boat Operations Area Boarding Frequency.



```
              US Coast Guard Group Monterey
                   Vessel Intelligence
          For the Period: 01/01/1980  - 01/01/1988
                *** Stolen Vessel List ***
          ==========================================

Date: 08/15/1987                                      Page:    1

                  Vessel                       Date of
                   Name                     Intelligence
              -----------------             ------------
              miss barbs                    02/05/1986

              miss cannibis                 05/18/1985

              miss cocaine                  06/01/1985

              miss cocaine                  05/20/1986

              miss everyday                 08/08/1985

              miss mariposa                 05/06/1986
```

Figure B.29 Stolen Vessel Listing.



```
                      US Coast Guard Group Monterey
                    Boarding Unsafe Condition Summary
                 For the Period: 01/01/1980  - 01/01/1988
                 ========================================

                     Total Boardings  -          8

Date: 08/15/1987                                          Page:   1

                                                   Percentage of
   Unsafe Condition          Occurrences         Total Boardings
---------------------        -----------         ---------------

Unsafe Conditions ---
Terminated Use ...........        3                  37.50 %

Corrected on the Spot .....       0                   0.00 %

Overloaded ................       2                  25.00 %

Accumulation of
Fuel in Bilges ...........        3                  37.50 %

Fuel Leak ---
Engine or Fuel System .....       2                  25.00 %

Manifestly Unsafe Voyage ..       3                  37.50 %

Hazardous Bars
(13th District) ...........       1                  12.50 %

Alcohol Condition .........       6                  75.00 %

Other .....................       1                  12.50 %

--------------------------------------------------------------

Total Unsafe Conditions ...      21

==============================================================

Warning Issued ............       4                  50.00 %

No Violation Found ........       0                   0.00 %
```

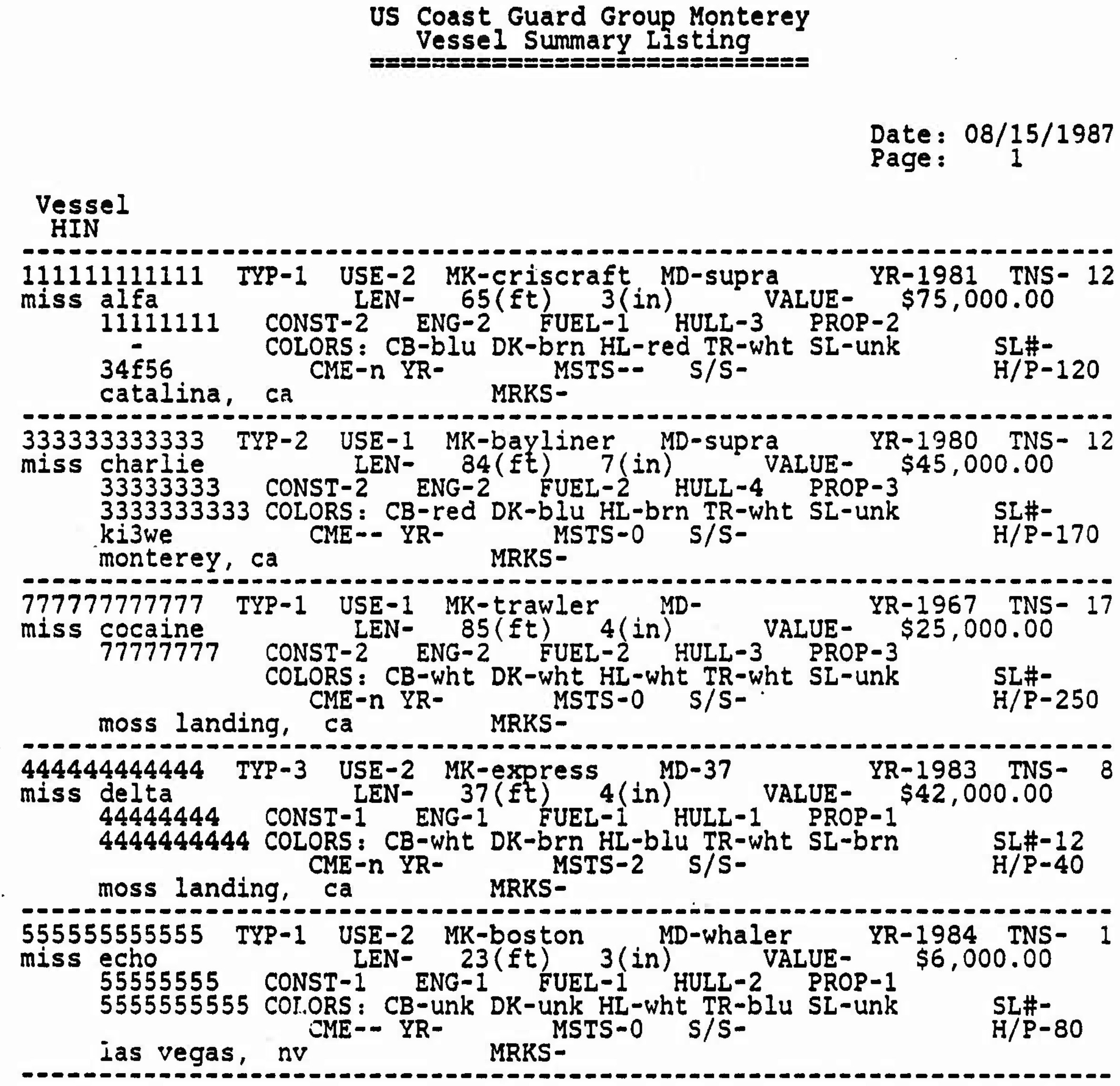
Figure B.30 Boarding Unsafe Condition Summary.

```
                      US Coast Guard Group Monterey
                   Demographics for Violation --- 54
                              Numbering
               For the Period: 01/01/1980  - 01/01/1988
                  *** Vessel Construction Type ***
               ==========================================

Date: 08/15/1987                                              Page:   1

                                         Percentage           Percentage
Construction                               Total                Total
    Type            Occurrences         Violation-54          Boardings
------------        -----------         ------------          ----------
open                          2             28.57 %              25.00 %

closed                        5             71.43 %              62.50 %

unknown                       0              0.00 %               0.00 %




-----------------------------------------------------------------------
Total Occurrences.            7                                  87.50 %
Total Boardings...            8
```

Figure B.31 Violation Summary by Vessel Construction Type.

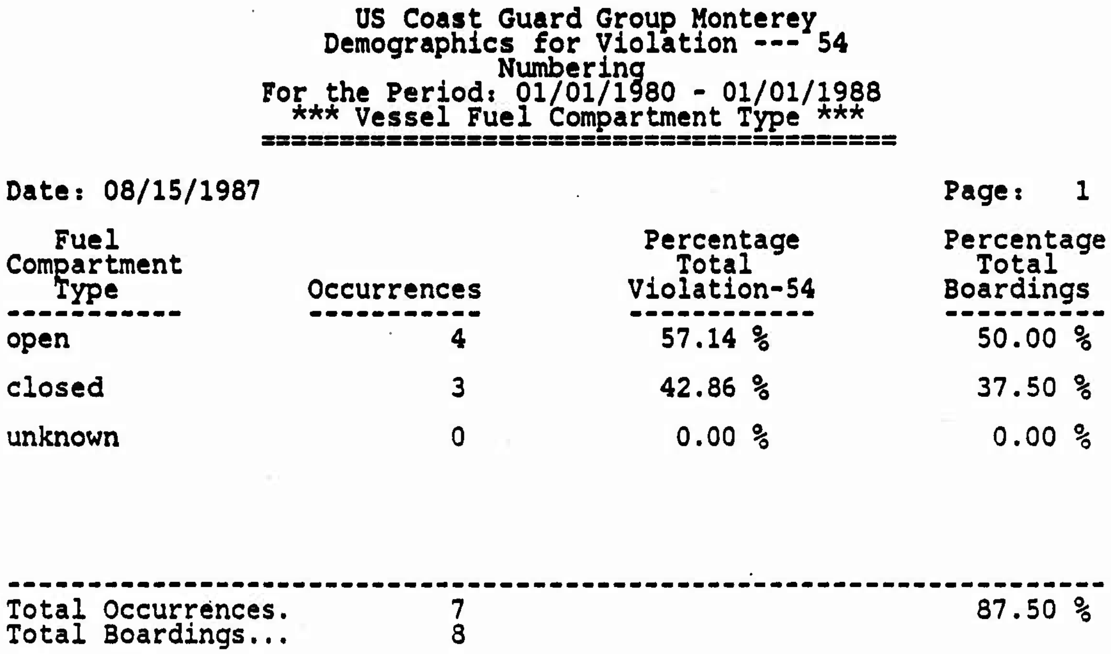
```
                        US Coast Guard Group Monterey
                     Demographics for Violation --- 54
                                Numbering
                 For the Period: 01/01/1980  - 01/01/1988
                 *** Vessel Engine Compartment Type ***
                 ==========================================

Date: 08/15/1987                                              Page:   1

  Engine                                 Percentage          Percentage
Compartment                                Total               Total
   Type             Occurrences         Violation-54         Boardings
-----------         -----------         ------------         ----------
open                          3             42.86 %              37.50 %

closed                        4             57.14 %              50.00 %

unknown                       0              0.00 %               0.00 %




-----------------------------------------------------------------------
Total Occurrences.            7                                  87.50 %
Total Boardings...            8
```

Figure B.32 Violation Summary by Engine Compartment Type.



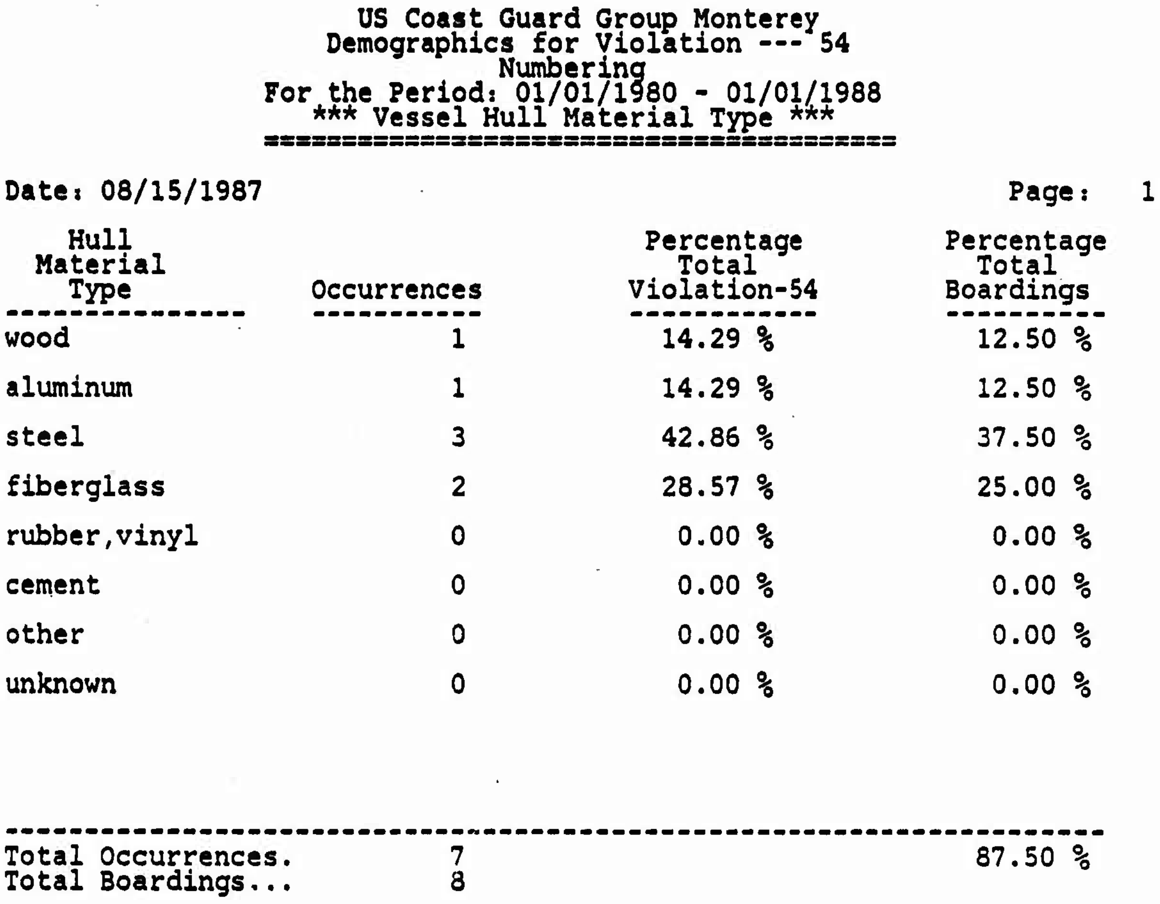
```
                        US Coast Guard Group Monterey
                           Vessel Summary Listing
                        ============================

                                                          Date: 08/15/1987
                                                          Page:    1
 Vessel
  HIN
----------------------------------------------------------------------
111111111111  TYP-1  USE-2  MK-criscraft  MD-supra      YR-1981  TNS- 12
miss alfa                LEN-   65(ft)   3(in)     VALUE-  $75,000.00
     11111111   CONST-2   ENG-2   FUEL-1   HULL-3   PROP-2
         -      COLORS: CB-blu DK-brn HL-red TR-wht SL-unk       SL#-
     34f56         CME-n YR-       MSTS--   S/S-                 H/P-120
     catalina,  ca              MRKS-
----------------------------------------------------------------------
333333333333  TYP-2  USE-1  MK-bayliner   MD-supra      YR-1980  TNS- 12
miss charlie             LEN-   84(ft)   7(in)     VALUE-  $45,000.00
     33333333   CONST-2   ENG-2   FUEL-2   HULL-4   PROP-3
     3333333333 COLORS: CB-red DK-blu HL-brn TR-wht SL-unk       SL#-
     ki3we         CME-- YR-       MSTS-0   S/S-                 H/P-170
     monterey, ca               MRKS-
----------------------------------------------------------------------
777777777777  TYP-1  USE-1  MK-trawler    MD-           YR-1967  TNS- 17
miss cocaine             LEN-   85(ft)   4(in)     VALUE-  $25,000.00
     77777777   CONST-2   ENG-2   FUEL-2   HULL-3   PROP-3
                COLORS: CB-wht DK-wht HL-wht TR-wht SL-unk       SL#-
                   CME-n YR-       MSTS-0   S/S-                 H/P-250
     moss landing,  ca          MRKS-
----------------------------------------------------------------------
444444444444  TYP-3  USE-2  MK-express    MD-37         YR-1983  TNS-  8
miss delta               LEN-   37(ft)   4(in)     VALUE-  $42,000.00
     44444444   CONST-1   ENG-1   FUEL-1   HULL-1   PROP-1
     4444444444 COLORS: CB-wht DK-brn HL-blu TR-wht SL-brn       SL#-12
                   CME-n YR-       MSTS-2   S/S-                 H/P-40
     moss landing,  ca          MRKS-
----------------------------------------------------------------------
555555555555  TYP-1  USE-2  MK-boston     MD-whaler     YR-1984  TNS-  1
miss echo                LEN-   23(ft)   3(in)     VALUE-   $6,000.00
     55555555   CONST-1   ENG-1   FUEL-1   HULL-2   PROP-1
     5555555555 COLORS: CB-unk DK-unk HL-wht TR-blu SL-unk       SL#-
                   CME-- YR-       MSTS-0   S/S-                 H/P-80
     las vegas,  nv             MRKS-
----------------------------------------------------------------------
```

Figure B.33 Vessel Listing.



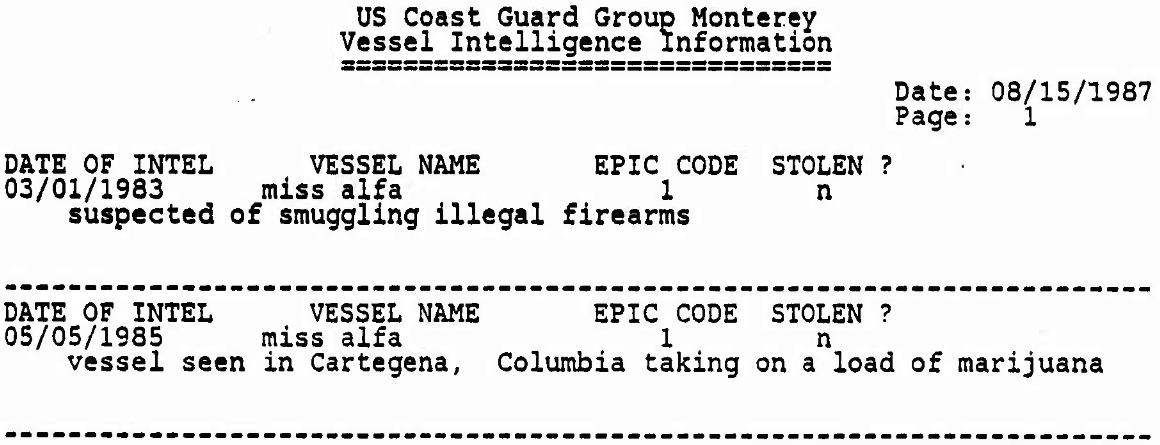
```
                 US Coast Guard Group Monterey
               Demographics for Violation --- 54
                          Numbering
              For the Period: 01/01/1980 - 01/01/1988
                *** Vessel Fuel Compartment Type ***
              ========================================

Date: 08/15/1987                                      Page:   1

    Fuel                                Percentage     Percentage
Compartment                                Total          Total
    Type           Occurrences         Violation-54     Boardings
-----------        -----------         ------------    ----------
open                         4            57.14 %        50.00 %

closed                       3            42.86 %        37.50 %

unknown                      0             0.00 %         0.00 %



------------------------------------------------------------------
Total Occurrences.           7                           87.50 %
Total Boardings...           8
```

Figure B.34 Violation Summary by Fuel Compartment Type.



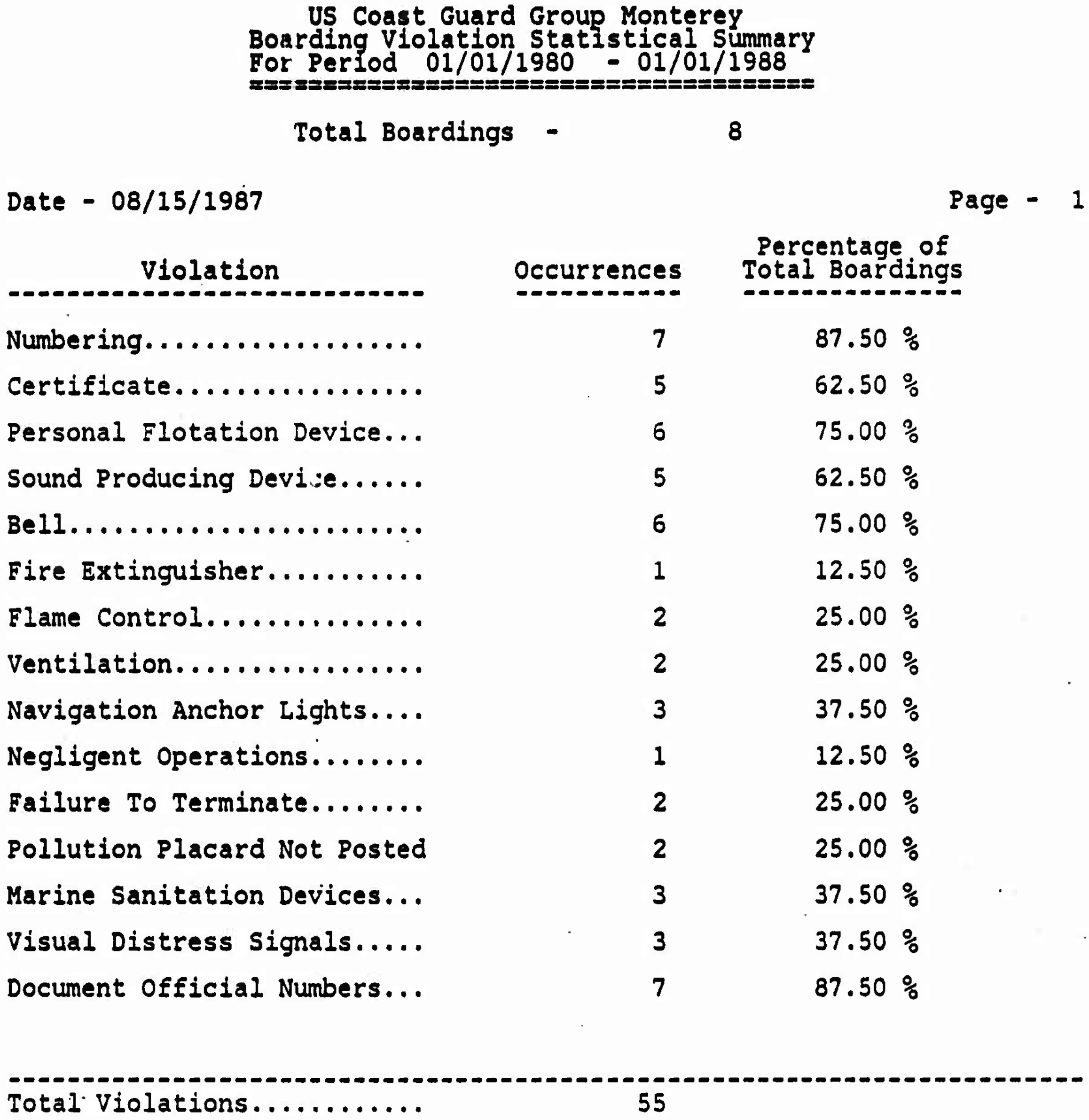
```
                    US Coast Guard Group Monterey
                   Demographics for Violation --- 54
                              Numbering
               For the Period: 01/01/1980 - 01/01/1988
                   *** Vessel Hull Material Type ***
               ========================================

Date: 08/15/1987                                              Page:   1

  Hull
 Material                                 Percentage        Percentage
  Type              Occurrences             Total             Total
                                         Violation-54       Boardings
---------------     -----------          ------------       ----------
wood                          1             14.29 %          12.50 %

aluminum                      1             14.29 %          12.50 %

steel                         3             42.86 %          37.50 %

fiberglass                    2             28.57 %          25.00 %

rubber,vinyl                  0              0.00 %           0.00 %

cement                        0              0.00 %           0.00 %

other                         0              0.00 %           0.00 %

unknown                       0              0.00 %           0.00 %



  --------------------------------------------------------------------
Total Occurrences.            7                              87.50 %
Total Boardings...            8
```

Figure B.35 Violation Summary by Hull Material Type.



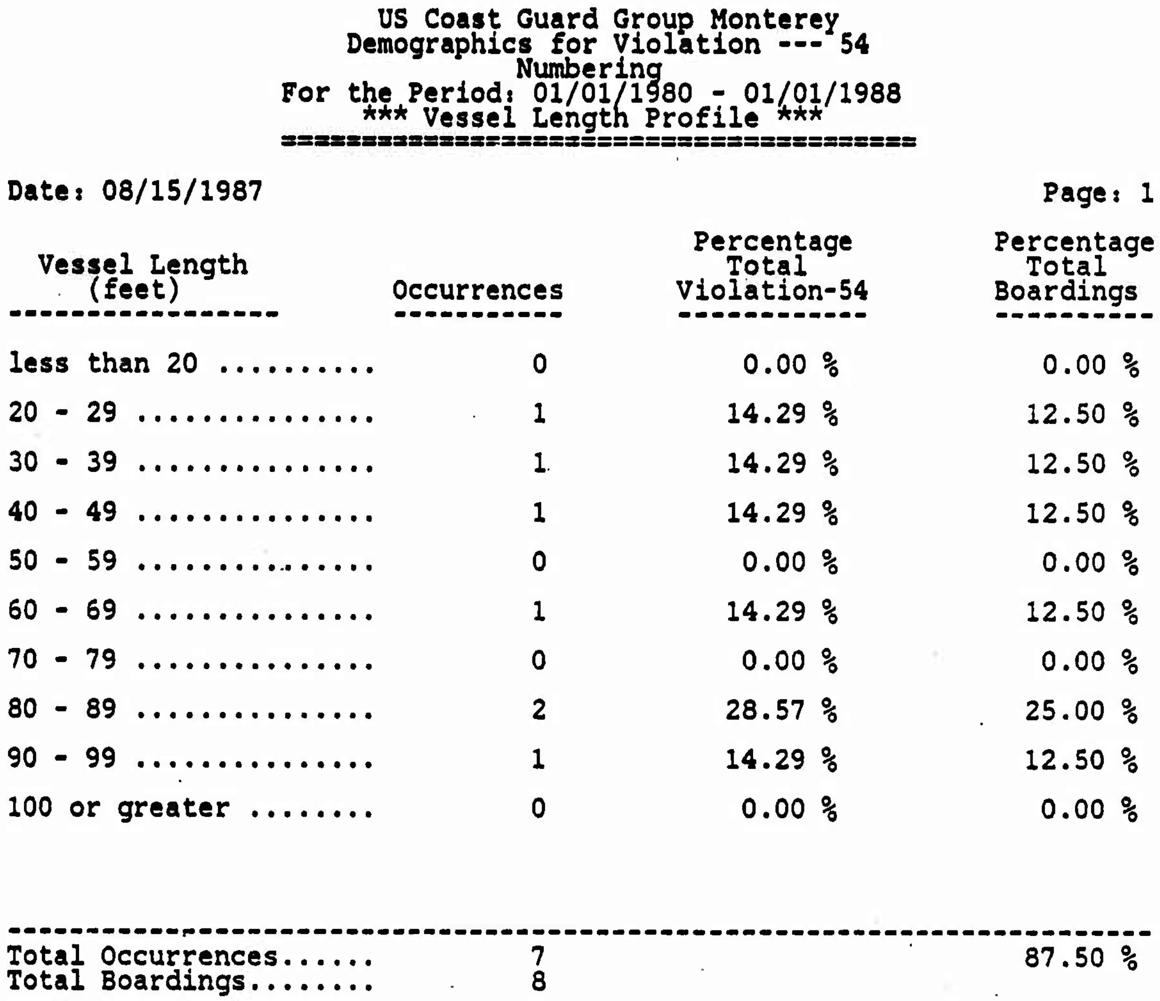
```
                    US Coast Guard Group Monterey
                   Vessel Intelligence Information
                   ===============================
                                                    Date: 08/15/1987
                                                    Page:    1

DATE OF INTEL       VESSEL NAME       EPIC CODE  STOLEN ?
03/01/1983       miss alfa                1         n
    suspected of smuggling illegal firearms

------------------------------------------------------------------------
DATE OF INTEL       VESSEL NAME       EPIC CODE  STOLEN ?
05/05/1985       miss alfa                1         n
    vessel seen in Cartegena,  Columbia taking on a load of marijuana

------------------------------------------------------------------------
```

Figure B.36 Query Vessel Intelligence History.



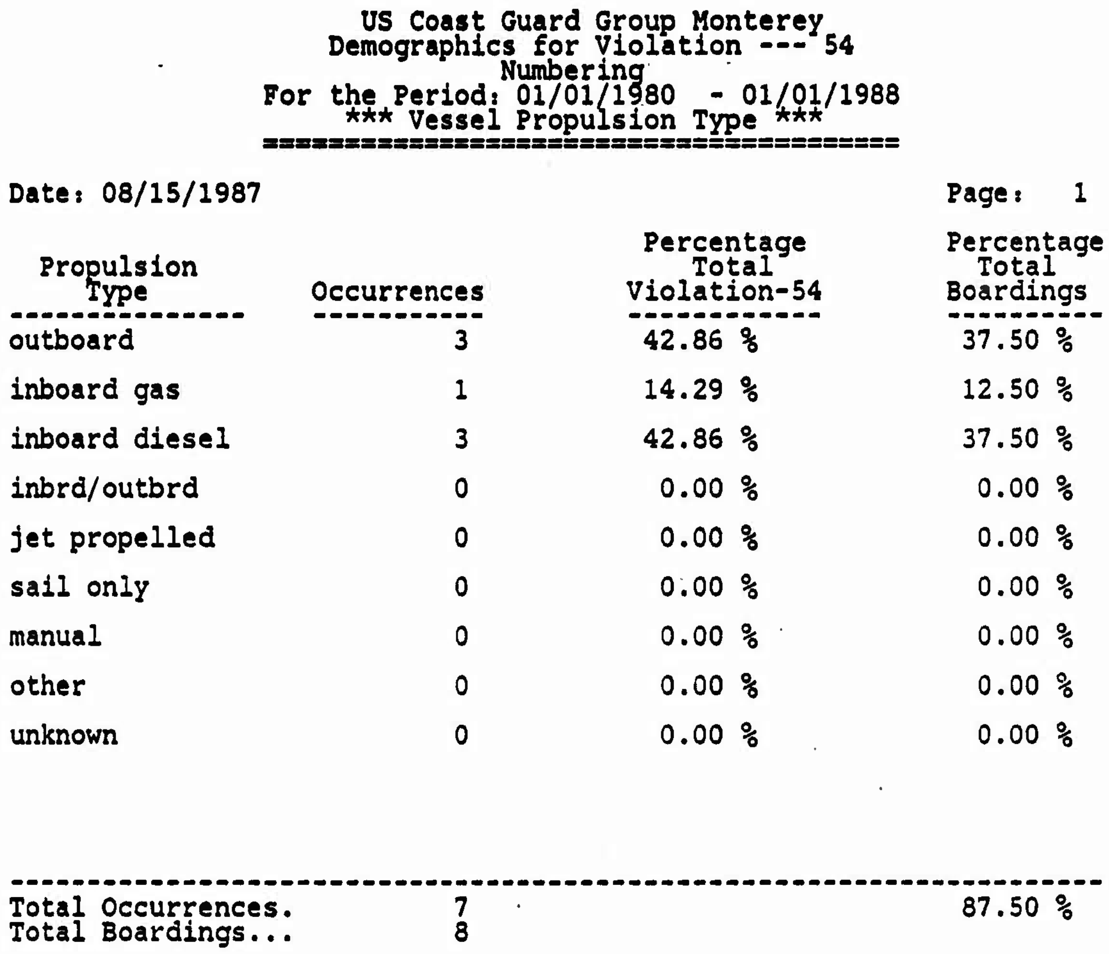
```
                   US Coast Guard Group Monterey
                Boarding Violation Statistical Summary
                For Period  01/01/1980   - 01/01/1988
                ======================================

                   Total Boardings  -         8

Date - 08/15/1987                                              Page -  1

                                                      Percentage of
         Violation                    Occurrences    Total Boardings
----------------------------          -----------    ---------------

Numbering...................                    7         87.50 %
Certificate.................                    5         62.50 %
Personal Flotation Device...                    6         75.00 %
Sound Producing Device......                    5         62.50 %
Bell........................                    6         75.00 %
Fire Extinguisher...........                    1         12.50 %
Flame Control...............                    2         25.00 %
Ventilation.................                    2         25.00 %
Navigation Anchor Lights....                    3         37.50 %
Negligent Operations........                    1         12.50 %
Failure To Terminate........                    2         25.00 %
Pollution Placard Not Posted                    2         25.00 %
Marine Sanitation Devices...                    3         37.50 %
Visual Distress Signals.....                    3         37.50 %
Document Official Numbers...                    7         87.50 %

--------------------------------------------------------------------------
Total Violations...........                    55
```

Figure B.37 Boarding Violation Summary.



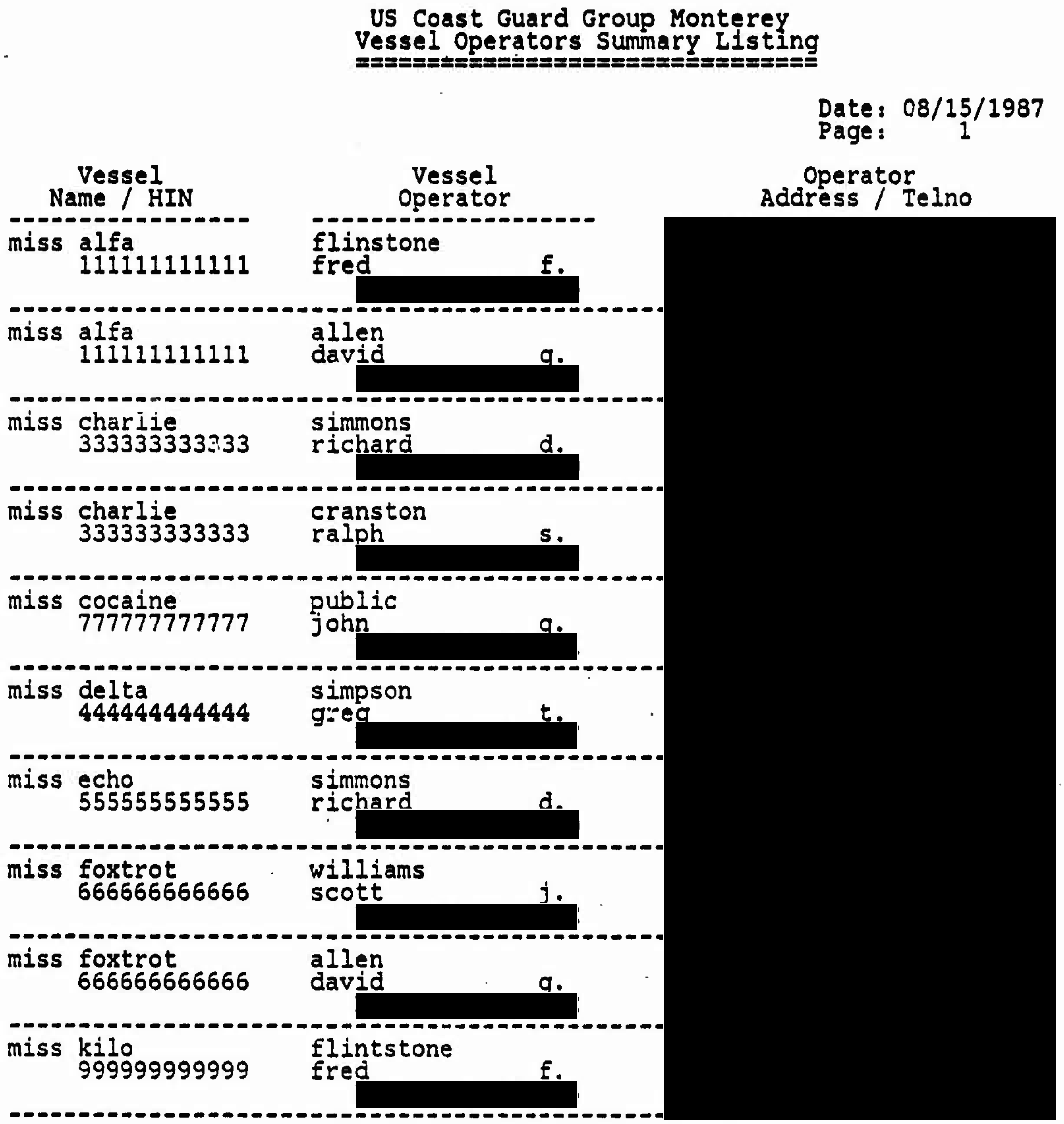
```
                   US Coast Guard Group Monterey
                 Demographics for Violation --- 54
                             Numbering
              For the Period: 01/01/1980 - 01/01/1988
                  *** Vessel Length Profile ***
              =======================================

Date: 08/15/1987                                                 Page: 1

                                            Percentage          Percentage
    Vessel Length                             Total               Total
       (feet)              Occurrences    Violation-54          Boardings
 ----------------          ----------     ------------          ----------

 less than 20 ..........        0            0.00 %               0.00 %

 20 - 29 ...............        1           14.29 %              12.50 %

 30 - 39 ...............        1           14.29 %              12.50 %

 40 - 49 ...............        1           14.29 %              12.50 %

 50 - 59 ...............        0            0.00 %               0.00 %

 60 - 69 ...............        1           14.29 %              12.50 %

 70 - 79 ...............        0            0.00 %               0.00 %

 80 - 89 ...............        2           28.57 %              25.00 %

 90 - 99 ...............        1           14.29 %              12.50 %

 100 or greater ........        0            0.00 %               0.00 %


 ----------------------------------------------------------------------
 Total Occurrences......        7                                87.50 %
 Total Boardings........        8
```

Figure B.38 Violation Summary by Vessel Length.



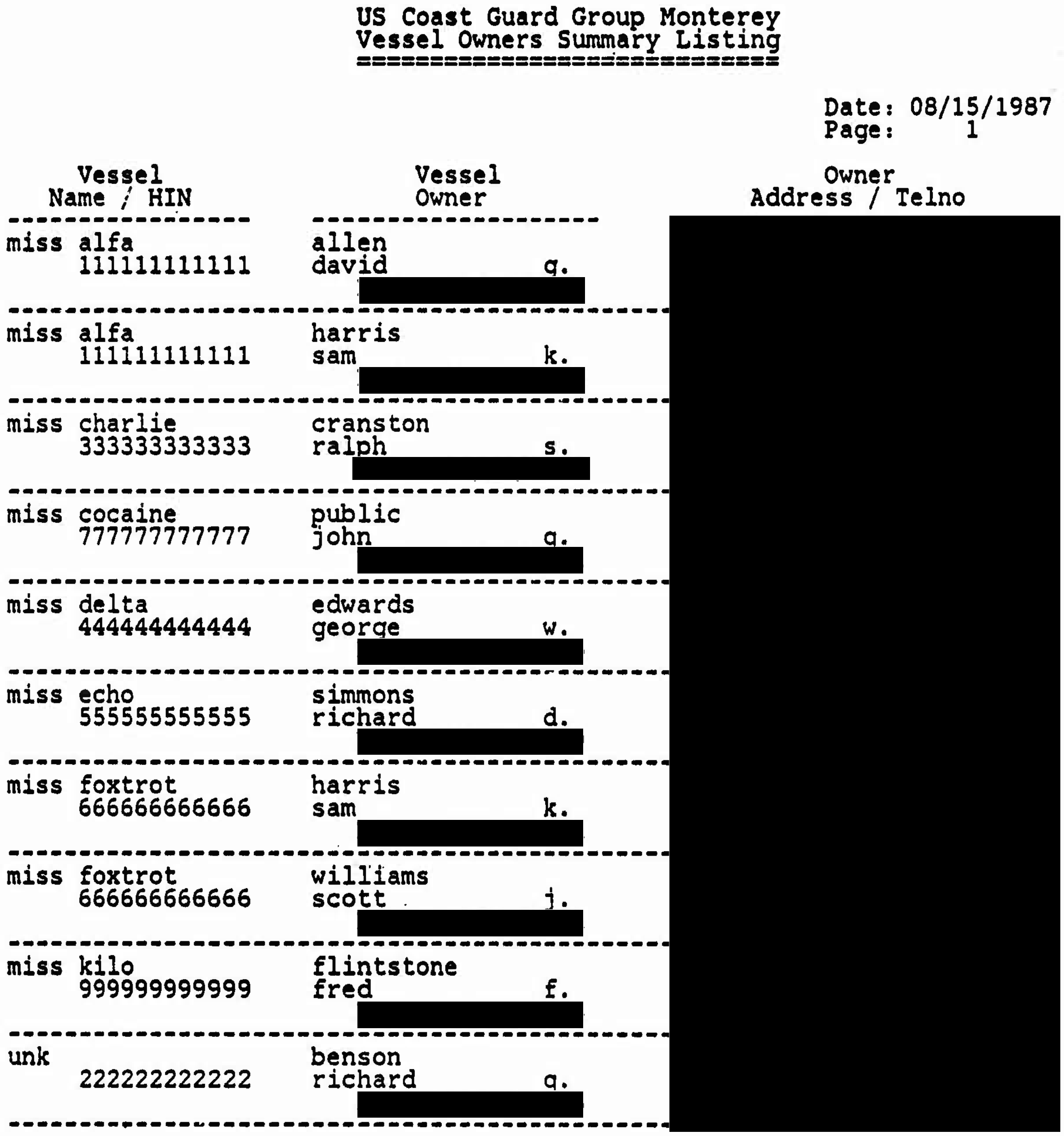
```
                    US Coast Guard Group Monterey
                 Demographics for Violation --- 54
                              Numbering
              For the Period: 01/01/1980  - 01/01/1988
                  *** Vessel Propulsion Type ***
              ==========================================

Date: 08/15/1987                                             Page:   1

                                       Percentage          Percentage
  Propulsion                             Total               Total
     Type           Occurrences       Violation-54         Boardings
---------------     -----------       ------------         ----------
outboard                      3        42.86 %              37.50 %

inboard gas                   1        14.29 %              12.50 %

inboard diesel                3        42.86 %              37.50 %

inbrd/outbrd                  0         0.00 %               0.00 %

jet propelled                 0         0.00 %               0.00 %

sail only                     0         0.00 %               0.00 %

manual                        0         0.00 %               0.00 %

other                         0         0.00 %               0.00 %

unknown                       0         0.00 %               0.00 %



-----------------------------------------------------------------------
Total Occurrences.            7                             87.50 %
Total Boardings...            8
```

Figure B.39 Violation Summary by Vessel Propulsion Type.



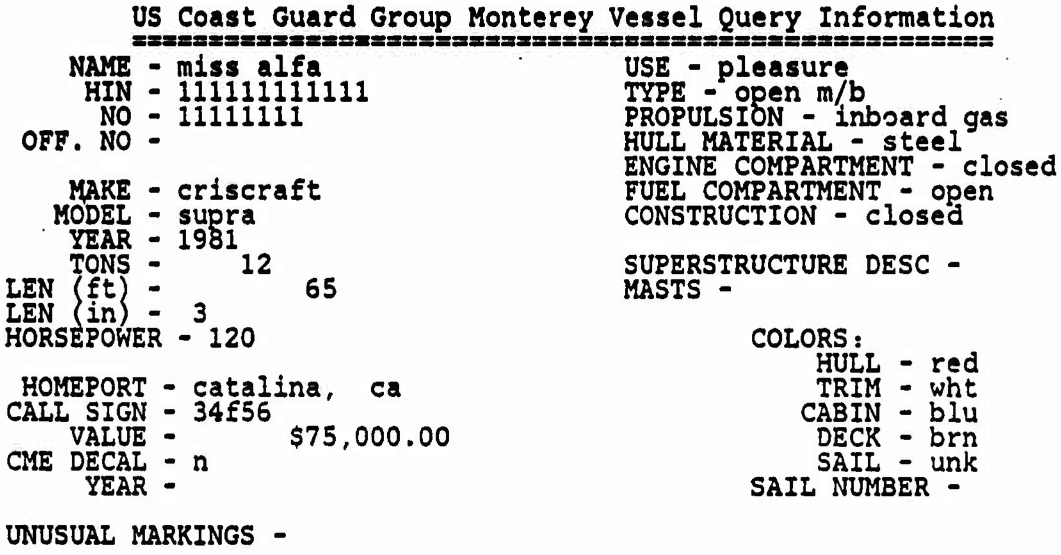
```
                         US Coast Guard Group Monterey
                        Vessel Operators Summary Listing
                        ================================

                                                         Date: 08/15/1987
                                                         Page:      1

      Vessel                 Vessel                       Operator
    Name / HIN              Operator                  Address / Telno
-----------------   ------------------
miss alfa           flinstone                         [redacted]
     111111111111   fred               f.
                        [redacted]
-------------------------------------------
miss alfa           allen
     111111111111   david              g.
                        [redacted]
-------------------------------------------
miss charlie        simmons
     333333333333   richard            d.
                        [redacted]
-------------------------------------------
miss charlie        cranston
     333333333333   ralph              s.
                        [redacted]
-------------------------------------------
miss cocaine        public
     777777777777   john               q.
                        [redacted]
-------------------------------------------
miss delta          simpson
     444444444444   greg               t.
                        [redacted]
-------------------------------------------
miss echo           simmons
     555555555555   richard            d.
                        [redacted]
-------------------------------------------
miss foxtrot        williams
     666666666666   scott              j.
                        [redacted]
-------------------------------------------
miss foxtrot        allen
     666666666666   david              g.
                        [redacted]
-------------------------------------------
miss kilo           flintstone
     999999999999   fred               f.
                        [redacted]
-------------------------------------------
```

Figure B.40 Vessel Operators Summary Listing.



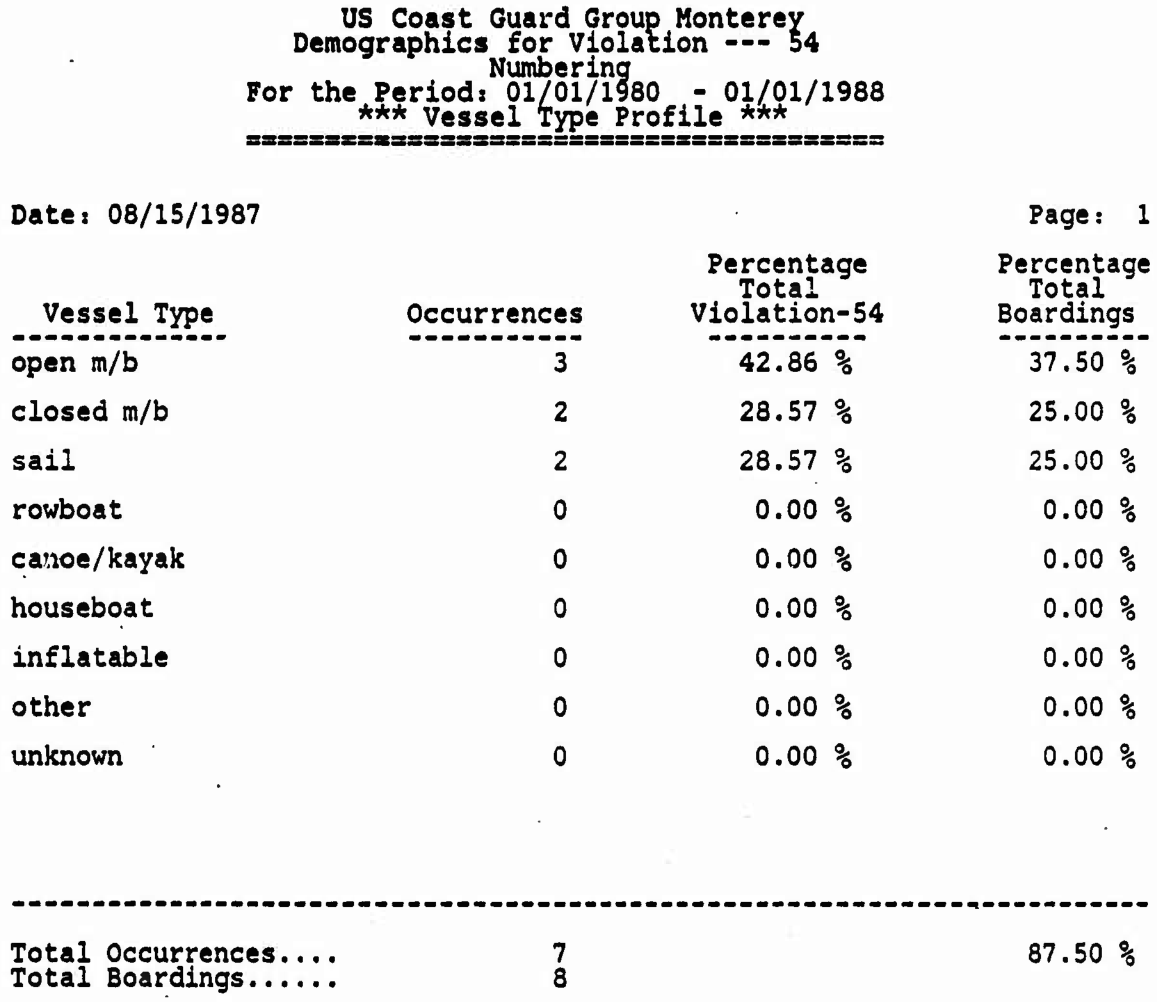
```
                         US Coast Guard Group Monterey
                         Vessel Owners Summary Listing
                         =============================

                                                            Date: 08/15/1987
                                                            Page:      1

     Vessel                     Vessel                        Owner
   Name / HIN                   Owner                   Address / Telno
----------------           -------------------
miss alfa                  allen                        [redacted]
     111111111111          david            g.
                             [redacted]
------------------------------------------------
miss alfa                  harris
     111111111111          sam              k.
                             [redacted]
------------------------------------------------
miss charlie               cranston
     333333333333          ralph            s.
                             [redacted]
------------------------------------------------
miss cocaine               public
     777777777777          john             q.
                             [redacted]
------------------------------------------------
miss delta                 edwards
     444444444444          george           w.
                             [redacted]
------------------------------------------------
miss echo                  simmons
     555555555555          richard          d.
                             [redacted]
------------------------------------------------
miss foxtrot               harris
     666666666666          sam              k.
                             [redacted]
------------------------------------------------
miss foxtrot               williams
     666666666666          scott            j.
                             [redacted]
------------------------------------------------
miss kilo                  flintstone
     999999999999          fred             f.
                             [redacted]
------------------------------------------------
unk                        benson
     222222222222          richard          g.
                             [redacted]
------------------------------------------------
```

Figure B.41 Vessel Owners Summary Listing.



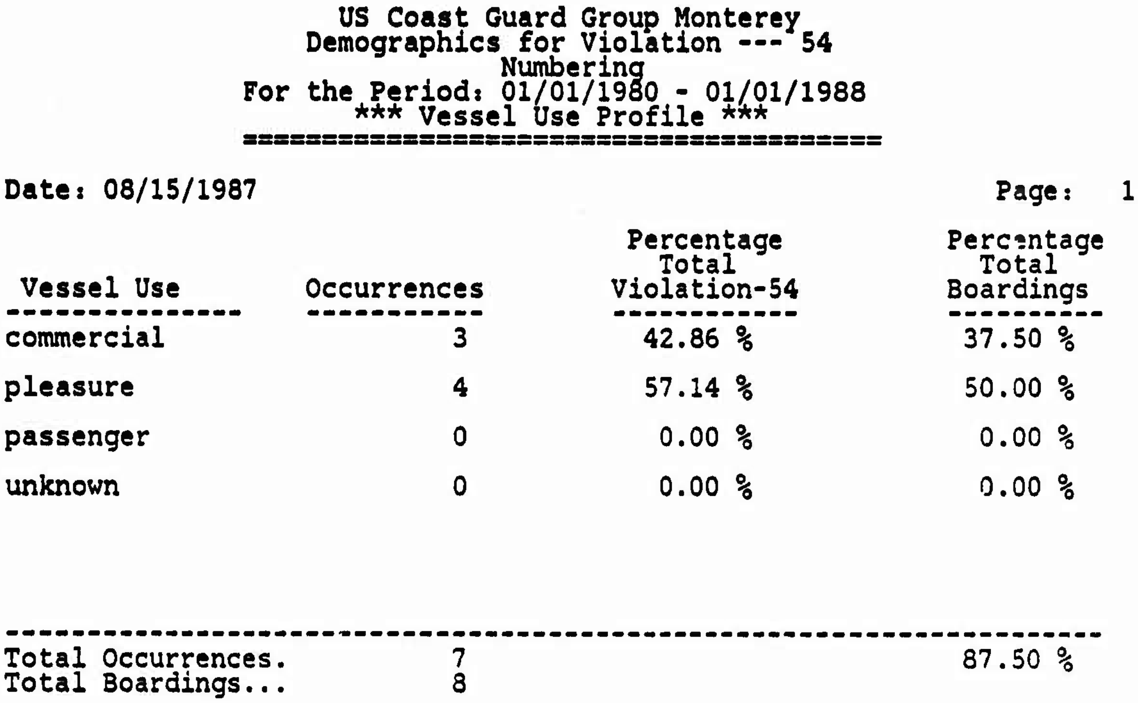
```
              US Coast Guard Group Monterey Vessel Query Information
              ======================================================
      NAME - miss alfa                    USE - pleasure
       HIN - 111111111111                 TYPE - open m/b
        NO - 11111111                     PROPULSION - inboard gas
   OFF. NO -                              HULL MATERIAL - steel
                                          ENGINE COMPARTMENT - closed
      MAKE - criscraft                    FUEL COMPARTMENT - open
     MODEL - supra                        CONSTRUCTION - closed
      YEAR - 1981
      TONS -     12                       SUPERSTRUCTURE DESC -
  LEN (ft) -         65                   MASTS -
  LEN (in) -  3
  HORSEPOWER - 120                                COLORS:
                                                      HULL - red
   HOMEPORT - catalina,  ca                           TRIM - wht
  CALL SIGN - 34f56                                  CABIN - blu
      VALUE -        $75,000.00                       DECK - brn
  CME DECAL - n                                       SAIL - unk
       YEAR -                                     SAIL NUMBER -

  UNUSUAL MARKINGS -
```

Figure B.42 Query Vessel Information Listing.



```
                    US Coast Guard Group Monterey
               Demographics for Violation --- 54
                           Numbering
              For the Period: 01/01/1980  - 01/01/1988
                    *** Vessel Type Profile ***
              =======================================

Date: 08/15/1987                                                Page:  1

                                          Percentage         Percentage
                                            Total              Total
  Vessel Type           Occurrences      Violation-54        Boardings
--------------          -----------       ----------         ----------
open m/b                          3         42.86 %            37.50 %

closed m/b                        2         28.57 %            25.00 %

sail                              2         28.57 %            25.00 %

rowboat                           0          0.00 %             0.00 %

canoe/kayak                       0          0.00 %             0.00 %

houseboat                         0          0.00 %             0.00 %

inflatable                        0          0.00 %             0.00 %

other                             0          0.00 %             0.00 %

unknown                           0          0.00 %             0.00 %



-----------------------------------------------------------------------

Total Occurrences....             7                            87.50 %
Total Boardings......             8
```

Figure B.43 Violation Summary by Vessel Type.



```
                      US Coast Guard Group Monterey
                    Demographics for Violation --- 54
                                Numbering
                For the Period: 01/01/1980 - 01/01/1988
                        *** Vessel Use Profile ***
                ========================================

Date: 08/15/1987                                                Page:   1

                                      Percentage          Percentage
                                        Total               Total
 Vessel Use       Occurrences       Violation-54          Boardings
---------------   -----------       ------------          ----------
commercial                  3          42.86 %              37.50 %

pleasure                    4          57.14 %              50.00 %

passenger                   0           0.00 %               0.00 %

unknown                     0           0.00 %               0.00 %



---------------------------------------------------------------------
Total Occurrences.          7                                87.50 %
Total Boardings...          8
```

Figure B.44 Violation Summary by Vessel Use.

```
                  US Coast Guard Group Monterey
                Demographics for Violation --- 54
                            Numbering
              For the Period: 01/01/1980 - 01/01/1988
                    *** Vessel Year Profile ***
              ========================================

Date: 08/15/1987                                                Page:  1

                                                Percentage        Percentage
      Vessel                                       Total             Total
      Years                   Occurrences      Violation-54       Boardings
------------------            -----------      ------------       ----------

earlier than 1960 .....             1             14.29 %           12.50 %

1960 - 1964 ...........             0              0.00 %            0.00 %

1965 - 1969 ...........             1             14.29 %           12.50 %

1970 - 1974 ...........             0              0.00 %            0.00 %

1975 - 1979 ...........             0              0.00 %            0.00 %

1980 - 1984 ...........             5             71.43 %           62.50 %

1985 - 1989 ...........             0              0.00 %            0.00 %

1990 - 1994 ...........             0              0.00 %            0.00 %

1995 - 1999 ...........             0              0.00 %            0.00 %

2000 or later .........             0              0.00 %            0.00 %



--------------------------------------------------------------------------
Total Occurrences......             7                                0.00 %
Total Boardings........             8
```

Figure B.45 Violation Summary by Vessel Year.

TABLE 3

FIGURE - PROGRAM CORRESPONDENCE

FIGURE NUMBER	REPORT NAME
B.1	boarding
B.2	bomajor
B.3	bominor
B.4	boqprn
B.5	cmerpt
B.6	daynight
B.7	epicrpt
B.8	epicsum
B.9	immedsar
B.10	intelsum
B.11	operator
B.12	opervsls
B.13	opqprn
B.14	owners
B.15	ownvsls
B.16	owqprn
B.17	perintel
B.18	perqprn
B.19	pintqprn
B.20	sar
B.21	sardates
B.22	sardays
B.23	sarmajor
B.24	sarminor
B.46	sarqprn
B.26	sarsmbt
B.27	sartimes
B.28	smboatrpt
B.29	stolesum
B.30	unsfreq
B.31	vconstmo
B.32	vengmo
B.33	vessel
B.34	vfuelmo
B.35	vhullmo
B.36	vintqprn
B.37	violfreq
B.38	vlenftmo
B.39	vpropmo
B.40	vsloper
B.41	vslowner
B.42	vslqprn
B.43	vtypemo
B.44	vusemo
B.45	vyearmo

APPENDIX C
SARELTIS PHYSICAL STRUCTURE

The physical structure of SARELTIS is presented in this appendix through various types of database documentation. This documentation includes the following:

- Menu Hierarchy Charts (Figure C.1 through Figure C.11)
- A listing of all current Tables
- A listing of all current Report Formats
- A listing of all current Entry/Edit Forms
- Database Table Compositions
- A listing of Reports and Associated Tables
- A listing of Database Attributes sorted by Attribute Name
- A listing of Database Attributes sorted by Input Source
- A listing of Database Attributes sorted by Program Where Used
- A listing of Variables sorted by Variable Name
- A listing of Variables sorted by Program Where Used
- A listing of the Data Entry / Edit Rules

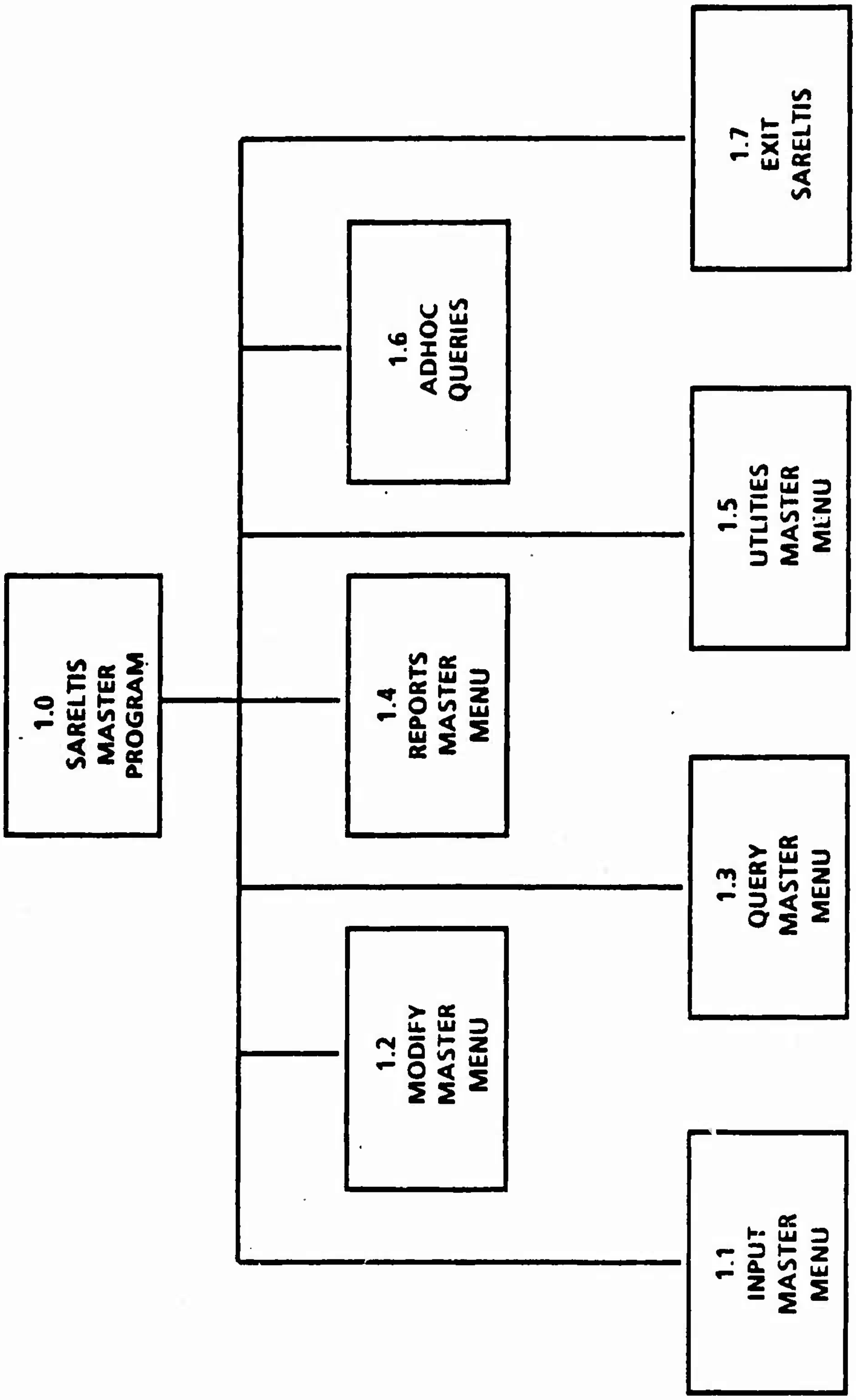


Figure C.1 SARELTIS Menu Hierarchy Chart.

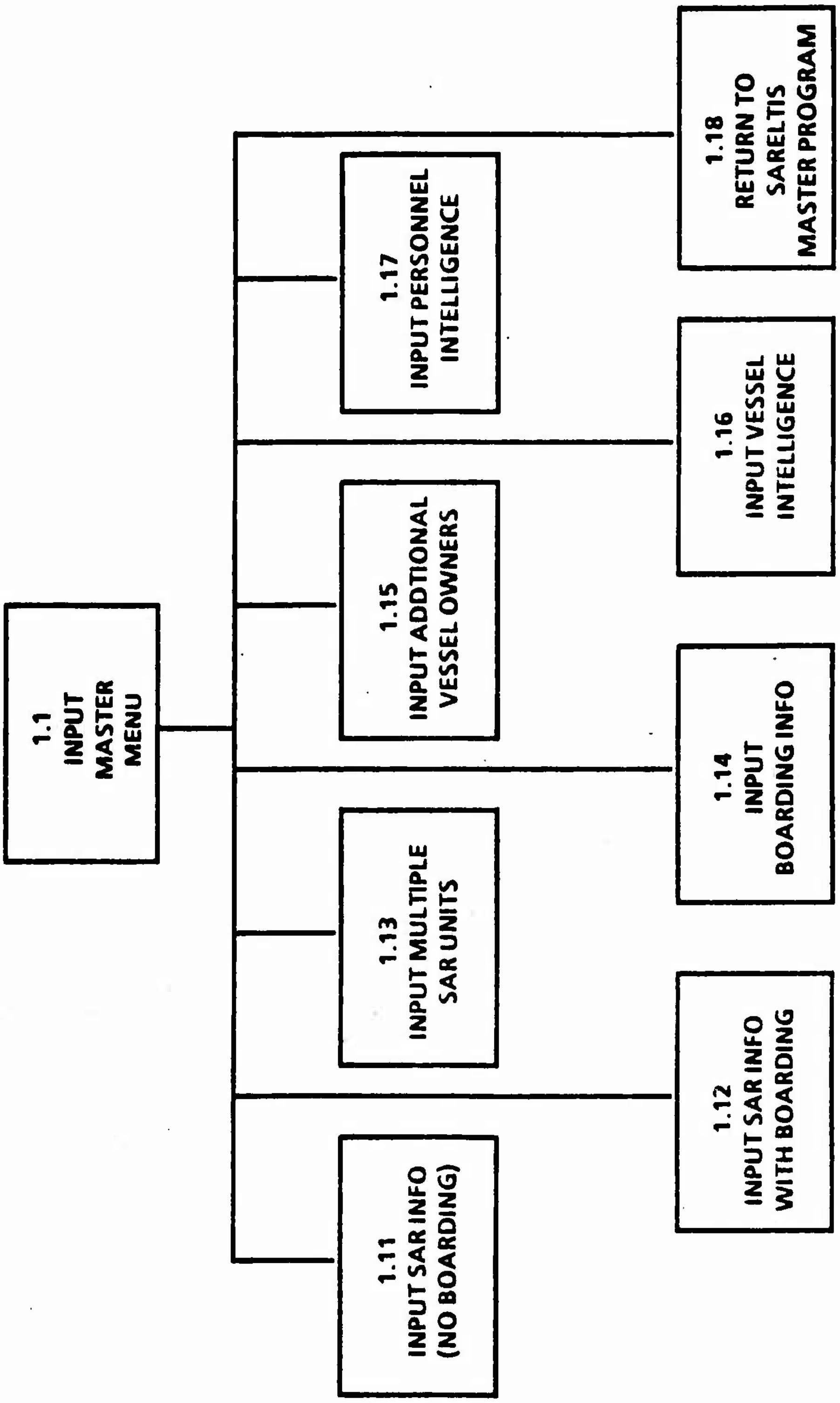


Figure C.2 Input Menu Hierarchy Chart.

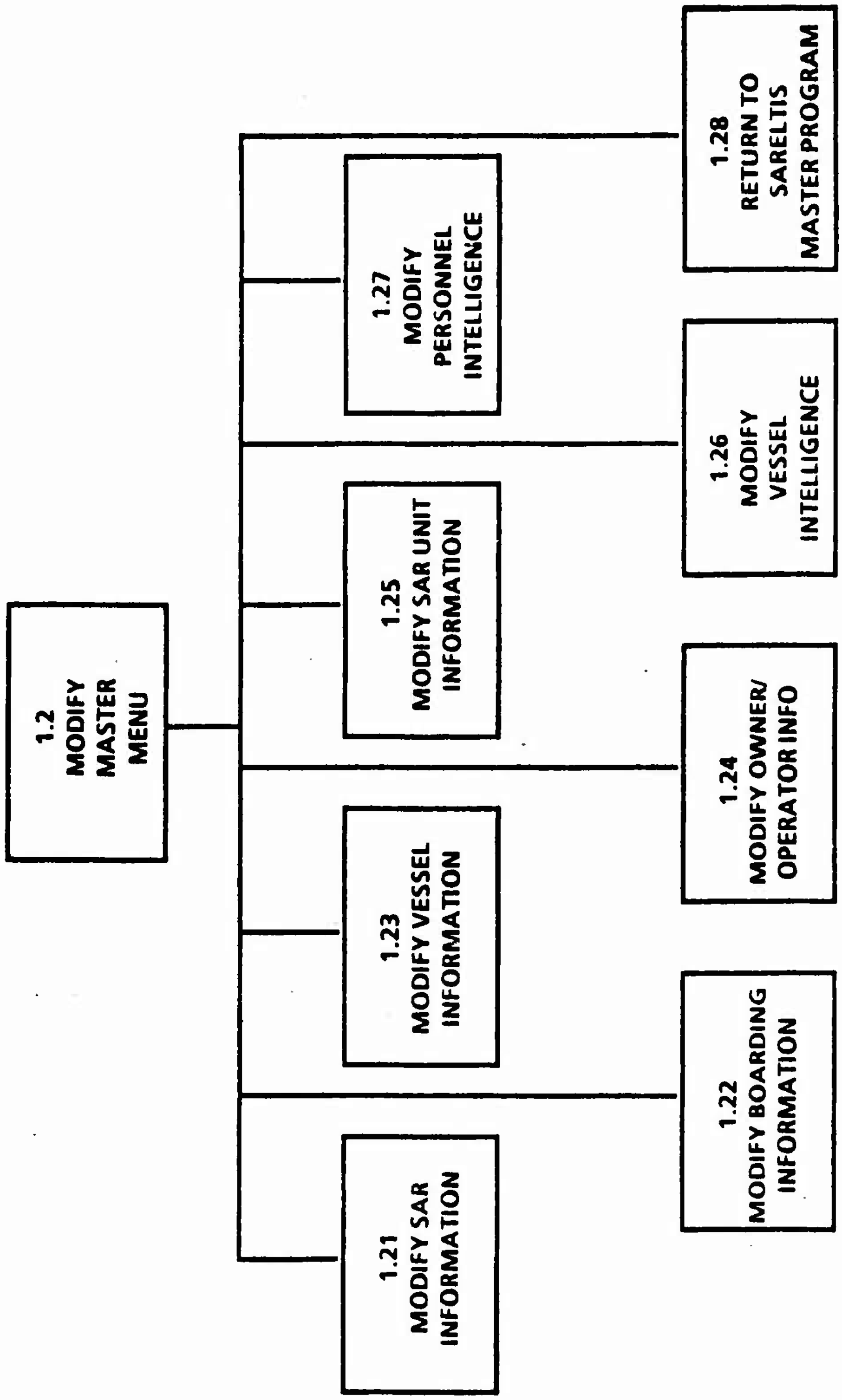

1.2 MODIFY MASTER MENU
1.21 MODIFY SAR INFORMATION
1.22 MODIFY BOARDING INFORMATION
1.23 MODIFY VESSEL INFORMATION
1.24 MODIFY OWNER/ OPERATOR INFO
1.25 MODIFY SAR UNIT INFORMATION
1.26 MODIFY VESSEL INTELLIGENCE
1.27 MODIFY PERSONNEL INTELLIGENCE
1.28 RETURN TO SARELTIS MASTER PROGRAM

Figure C.3 Modify Menu Hierarchy Chart.

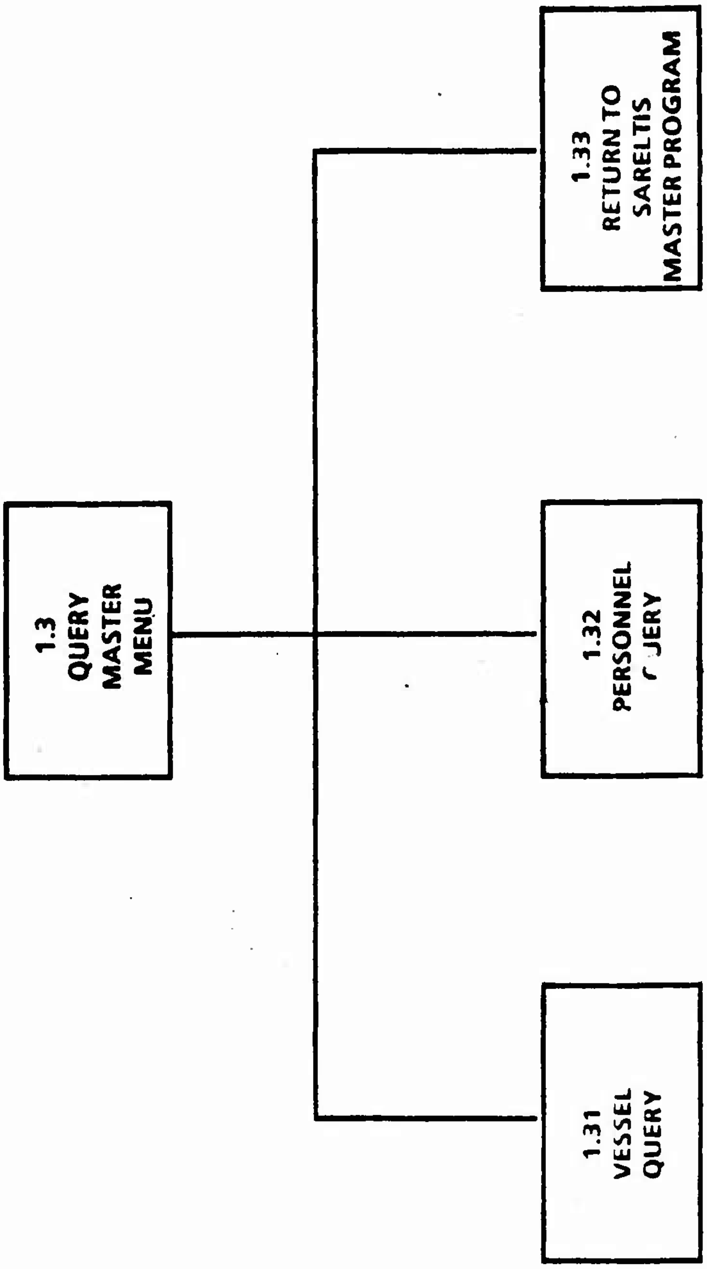


Figure C.4 Query Menu Hierarchy Chart.

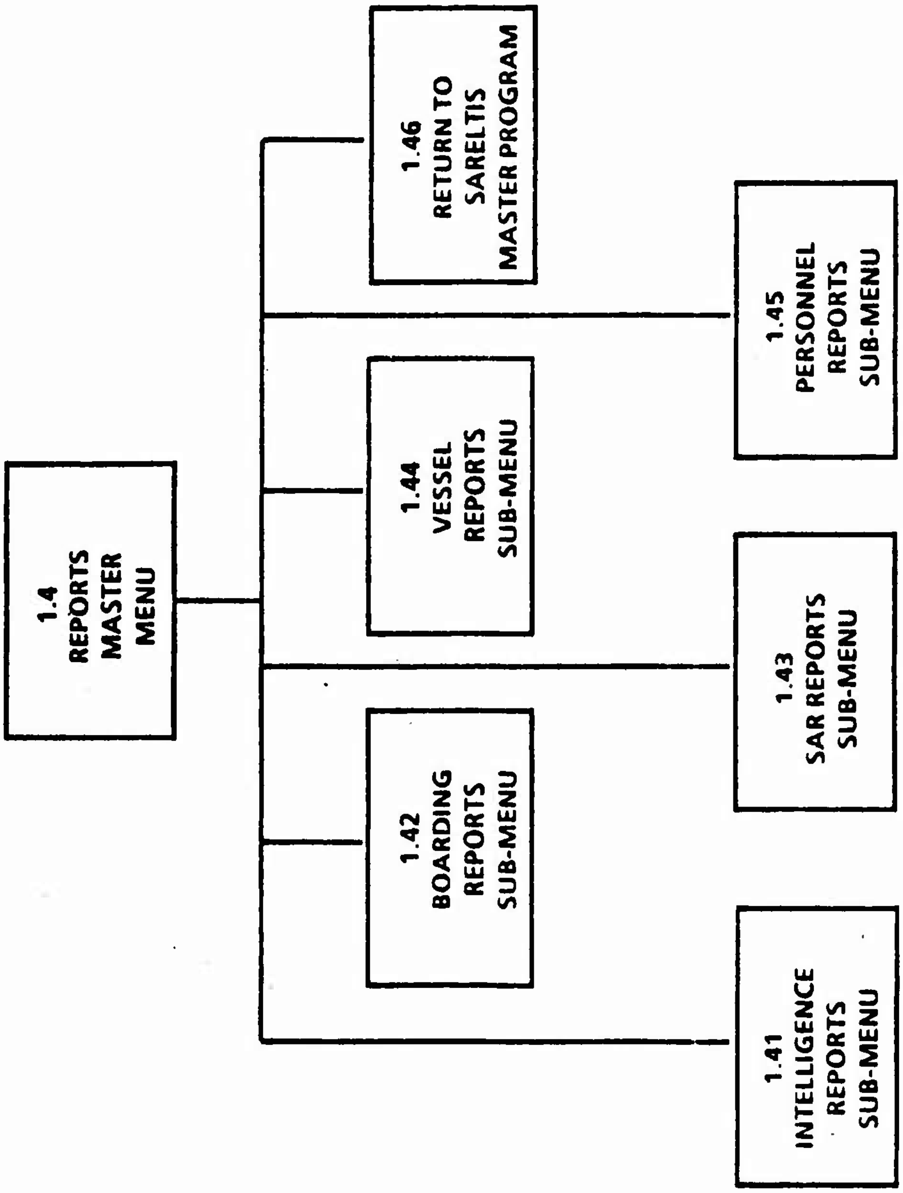

1.4 REPORTS MASTER MENU
1.41 INTELLIGENCE REPORTS SUB-MENU
1.42 BOARDING REPORTS SUB-MENU
1.43 SAR REPORTS SUB-MENU
1.44 VESSEL REPORTS SUB-MENU
1.45 PERSONNEL REPORTS SUB-MENU
1.46 RETURN TO SARELTIS MASTER PROGRAM

Figure C.5 Reports Master Menu Hierarchy Chart.

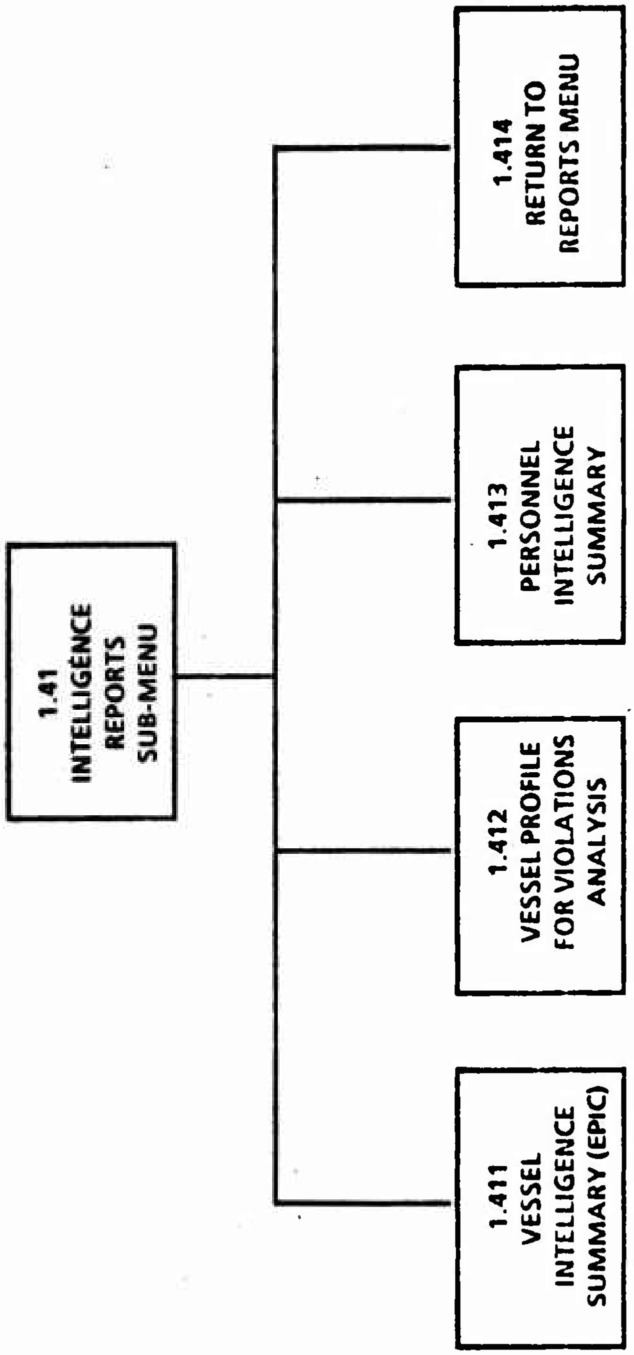


Figure C.6 Intelligence Reports Menu Hierarchy Chart.

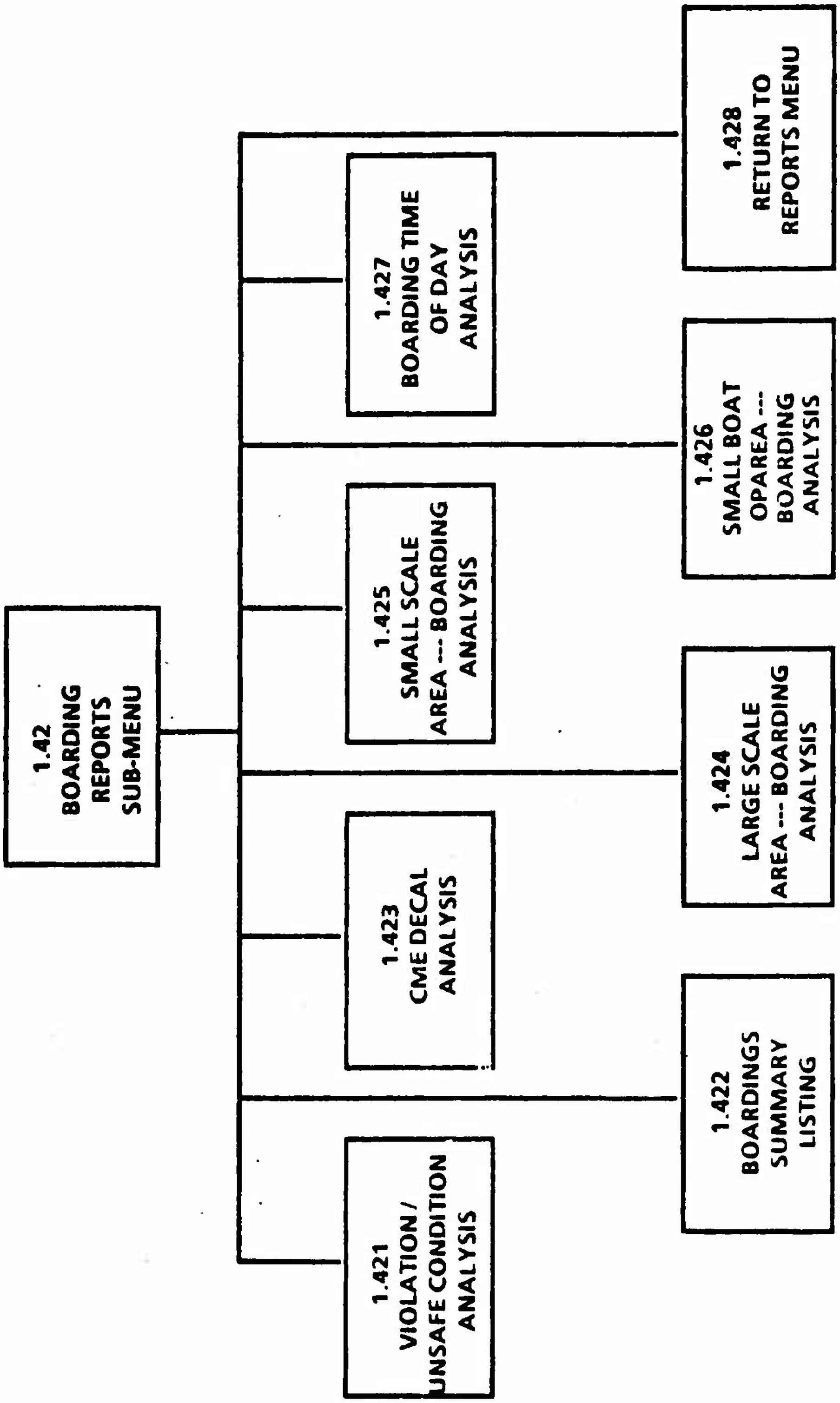

1.42 BOARDING REPORTS SUB-MENU
1.421 VIOLATION / UNSAFE CONDITION ANALYSIS
1.422 BOARDINGS SUMMARY LISTING
1.423 CME DECAL ANALYSIS
1.424 LARGE SCALE AREA --- BOARDING ANALYSIS
1.425 SMALL SCALE AREA --- BOARDING ANALYSIS
1.426 SMALL BOAT OPAREA --- BOARDING ANALYSIS
1.427 BOARDING TIME OF DAY ANALYSIS
1.428 RETURN TO REPORTS MENU

Figure C.7 Boarding Reports Menu Hierarchy Chart.

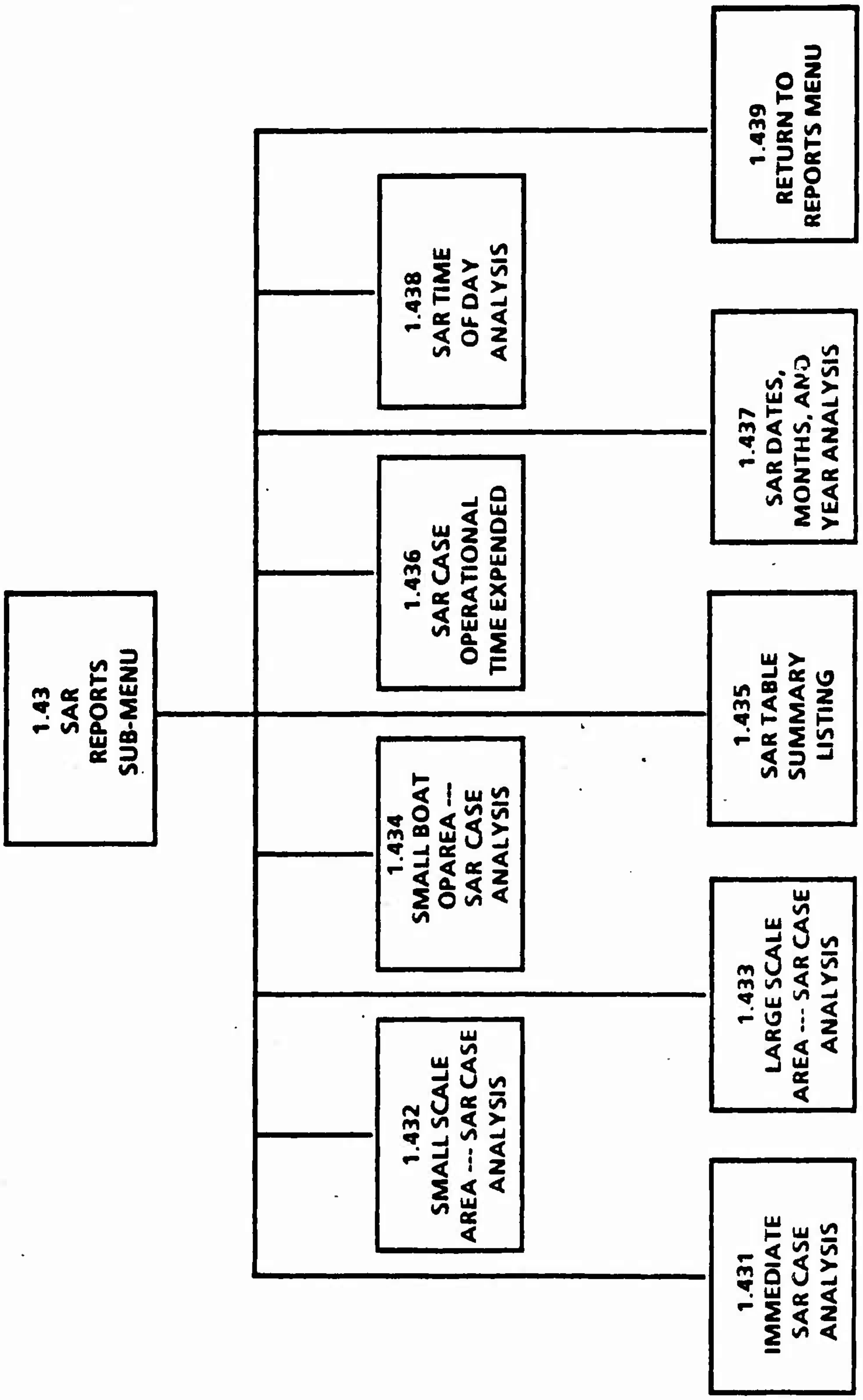

1.43 SAR REPORTS SUB-MENU
1.431 IMMEDIATE SAR CASE ANALYSIS
1.432 SMALL SCALE AREA --- SAR CASE ANALYSIS
1.433 LARGE SCALE AREA --- SAR CASE ANALYSIS
1.434 SMALL BOAT OPAREA --- SAR CASE ANALYSIS
1.435 SAR TABLE SUMMARY LISTING
1.436 SAR CASE OPERATIONAL TIME EXPENDED
1.437 SAR DATES, MONTHS, AND YEAR ANALYSIS
1.438 SAR TIME OF DAY ANALYSIS
1.439 RETURN TO REPORTS MENU

Figure C.8 SAR Reports Menu Hierarchy Chart.

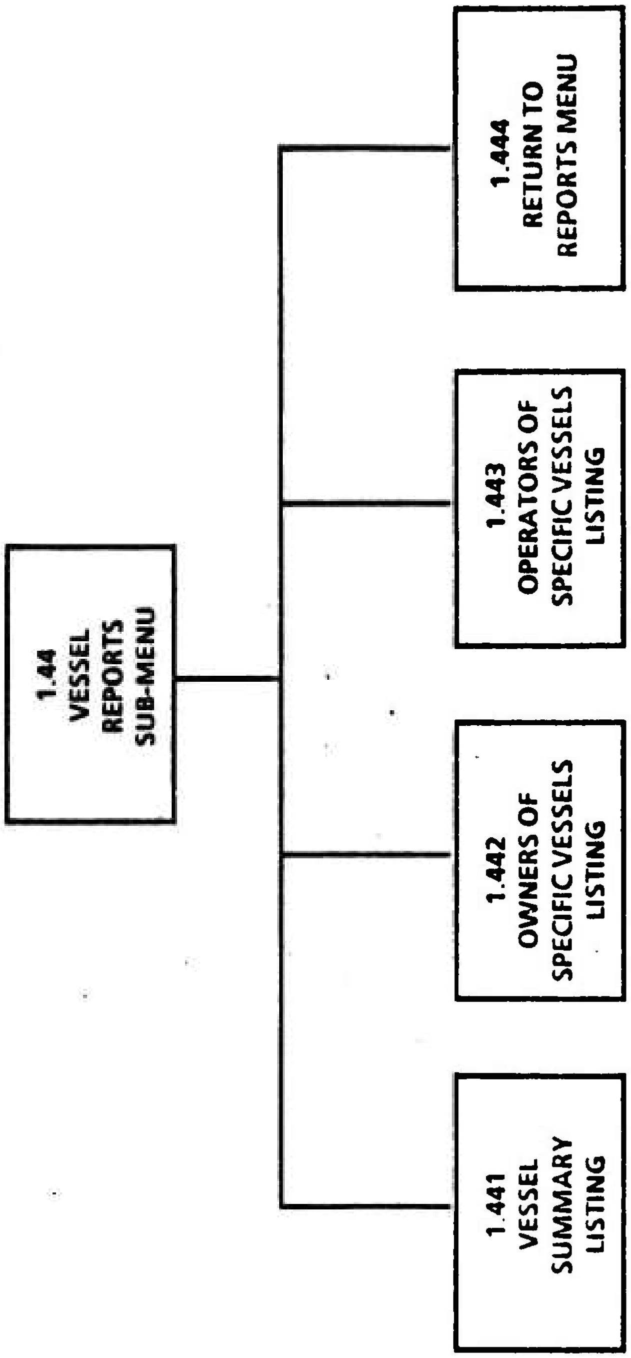


Figure C.9 Vessel Reports Menu Hierarchy Chart.

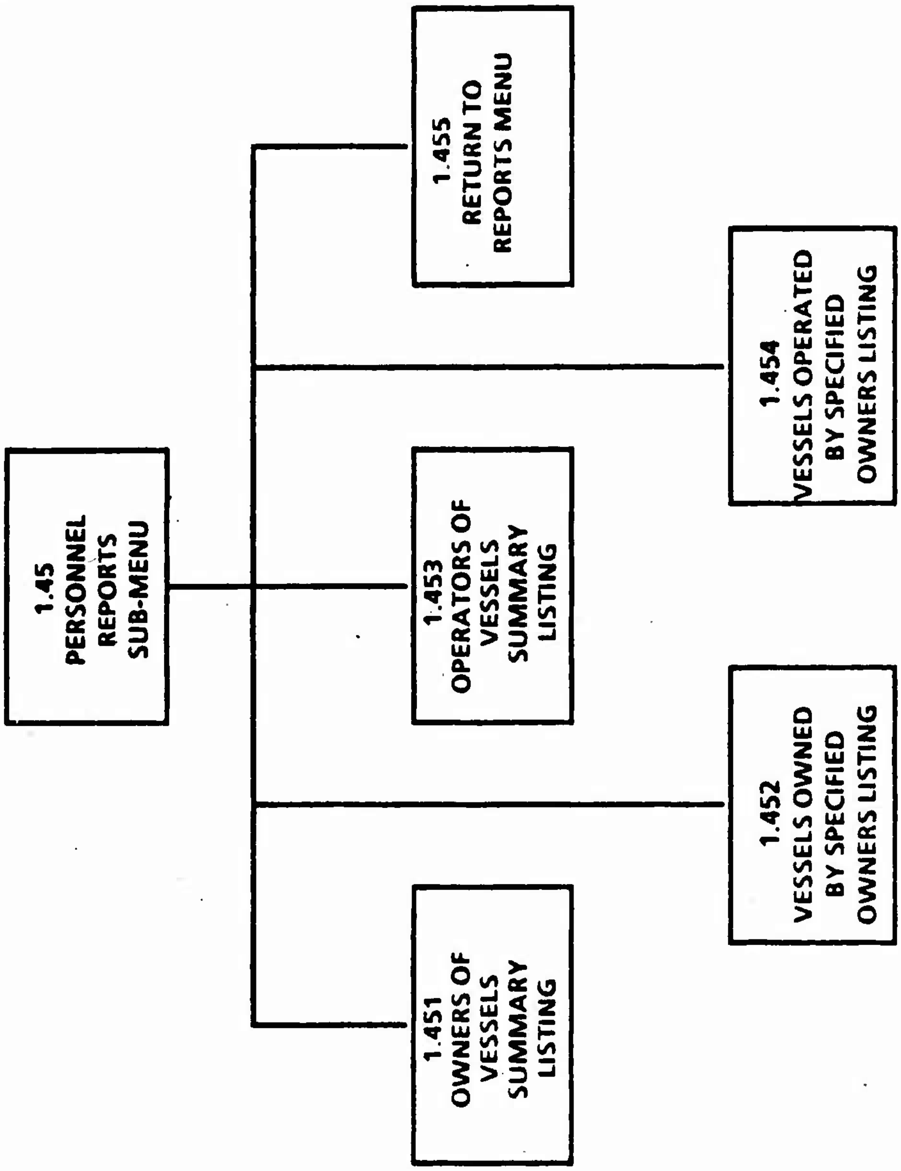


Figure C.10 Personnel Reports Menu Hierarchy Chart.

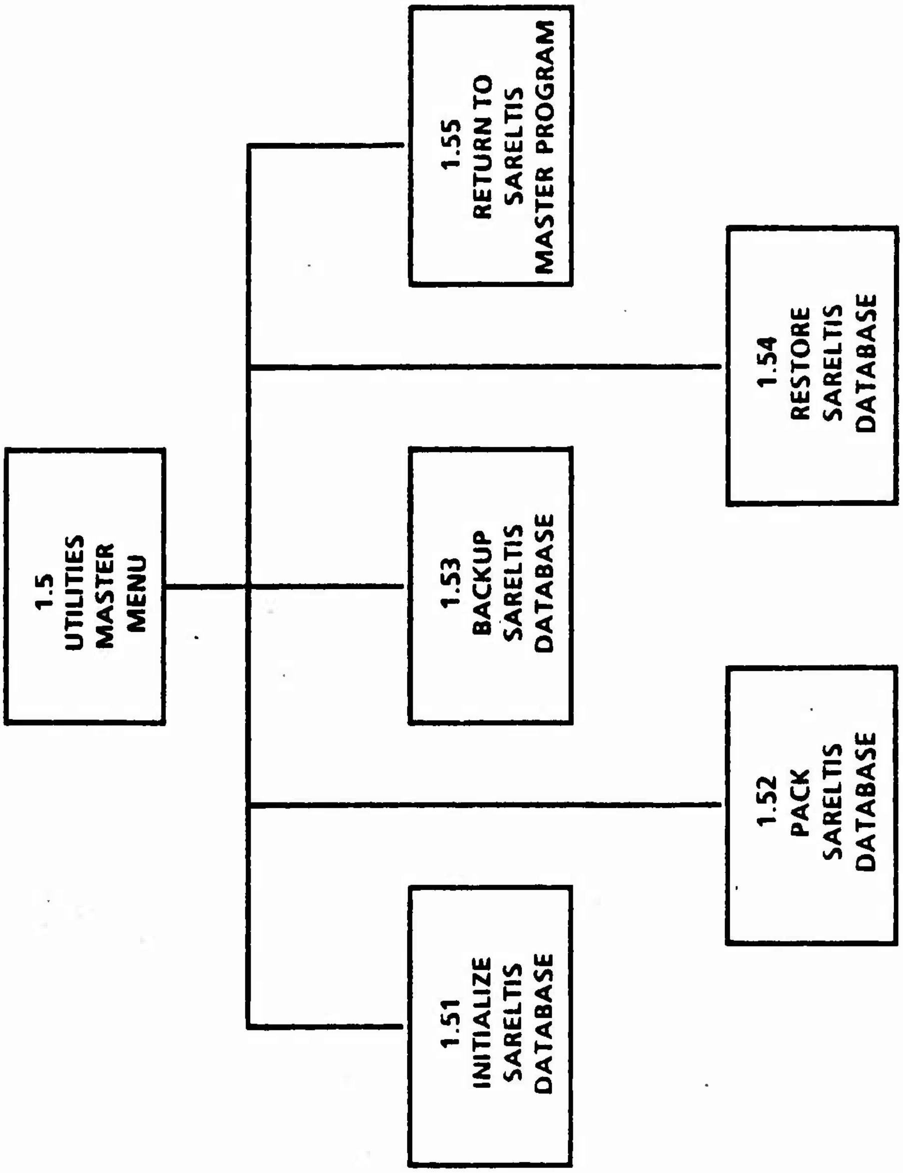


Figure C.11 Utilities Menu Hierarchy Chart.

US COAST GUARD GROUP MONTEREY
SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
INTELLIGENCE SYSTEM (SARELTIS) DATABASE

Tables

BOARDING
BOATPROP
BOATTYPE
BOATUSE
COLORS
CONSTR
COURSES
DATADICT
DAYS
DISTRICT
ENGCOMP
EPIC
FORMS
FUELCOMP
HULLMAT
MAJAREA
MINAREA
NATIONS
OPERATOR
OPOWN
OPVSL
OWNERS
OWVSL
PERINTEL
RULES
REPORTS
SAR
SARUNITS
STATES
STATUS
VARDICT
VESSEL
VIOLS
VSLINTEL

Reports

BOARDING
BOMAJOR
BOMINOR
BOQPRN
CMERPT
DATADICT
DAYNIGHT
EPICRPT
EPICSUM
IMMEDSAR
INTELSUM
OPERATOR
OPERVSLS
OPQPRN
OWNERS
OWNVSLS
OWQPRN
PERINTEL
PERQPRN
PINTQPRN
SAR
SARDATES
SARDAYS
SARMAJOR
SARMINOR
SARQPRN
SARSMBT
SARTIMES
SMBOATRP
STOLESUM
UNSFREQ
VARDICT
VCONSTMO
VENGMO
VESSEL
VFUELMO
VHULLMO
VINTQPRN
VIOLFREQ
VLENFTMO
VPROPMO
VSLOPER
VSLOWNER
VSLQPRN
VTYPEMO
VUSEMO
VYEARMO

Forms

BOMOD1
BOMOD2
BOMOD3
BRDQUERY
CG4100A1
CG4100A2
CG4100A3
CG4100A4
CG4100A5
CG4100A6
CG4100A7
CG4100A8
CG4100A9
DATADICT
HINQUERY
MAJAREA
MINAREA
OPERMOD1
OPERMOD2
OPQUERY
OWNQUERY
OWNRMOD1
OWNRMOD2
OWOPFORM
PERQUER1
PERQUER2
PINTFORM
PINTQUER
SARFORM1
SARFORM2
SARFORM3
SARFORM4
SARFORM5
SARFORM6
SARFORM7
SARFORM8
SARFORM9
SARMOD
SARQUERY
SBDFORM1
SUNITMOD
VARDICT
VHINFORM
VINTFORM
VINTQUER
VIOLS
VSLMOD1
VSLMOD2
VSLQUERY

```
Table: datadict
Read Password: NO
Modify Password: YES

Column definitions
# Name     Type      Length          Key
1 attname  TEXT        8 characters  yes
2 relname  TEXT        8 characters  yes
3 type     TEXT        7 characters
4 length   TEXT        3 characters
5 attdesc  TEXT       40 characters
6 source   TEXT       15 characters


Table: operator
Read Password: NO
Modify Password: NO

Column definitions
# Name      Type      Length          Key
1 oplname   TEXT       20 characters  yes
2 opfname   TEXT       15 characters  yes
3 opmi      TEXT        1 characters
4 opaddr    TEXT       25 characters
5 opcity    TEXT       18 characters
6 opstate   TEXT        2 characters
7 opzip     TEXT        9 characters
8 optelno   TEXT       12 characters
9 opdob     DATE        1 value(s)    yes
10 opcourse TEXT        1 characters
11 opid     INTEGER     1 value(s)    yes


Table: opvsl
Read Password: NO
Modify Password: NO

Column definitions
# Name     Type      Length          Key
1 vhinoper TEXT       12 characters  yes
2 idoper   INTEGER     1 value(s)    yes

Table: owners
Read Password: NO
Modify Password: NO

Column definitions
# Name      Type      Length          Key
1 ownlname  TEXT       20 characters  yes
2 ownfname  TEXT       15 characters  yes
3 ownmi     TEXT        1 characters
4 ownaddr   TEXT       25 characters
5 owncity   TEXT       18 characters
6 ownstate  TEXT        2 characters
7 ownzip    TEXT        9 characters
8 owntelno  TEXT       12 characters
9 owndob    DATE        1 value(s)    yes
10 ownid    INTEGER     1 value(s)    yes
```

```
Table: owvsl
Read Password: NO
Modify Password: NO

Column definitions
# Name     Type       Length          Key
1 vhinown  TEXT        12 characters  yes
2 idowner  INTEGER      1 value(s)    yes


Table: perintel
Read Password: NO
Modify Password: NO

Column definitions
# Name     Type       Length          Key
1 intlname TEXT        20 characters  yes
2 intdob   DATE         1 value(s)    yes
3 pintrem1 TEXT        70 characters
4 pintrem2 TEXT        70 characters
5 pintrem3 TEXT        70 characters
6 pintdate DATE         1 value(s)
7 intfname TEXT        15 characters
8 intmi    TEXT         1 characters


Table: sar
Read Password: NO
Modify Password: NO

Column definitions
# Name     Type       Length          Key
1 folderno TEXT         7 characters  yes
2 ucn      TEXT         8 characters  yes
3 mucn     TEXT         8 characters
4 sardate  DATE         1 value(s)    yes
5 sarday   TEXT         3 characters
6 sartime  TIME         1 value(s)
7 ssunrise TIME         1 value(s)
8 ssunset  TIME         1 value(s)
9 sarnod   TEXT        25 characters
10 sarpob  INTEGER      1 value(s)
11 sarvhin TEXT        12 characters  yes
12 sarlat  REAL         1 value(s)
13 sarlong REAL         1 value(s)
14 offshore INTEGER     1 value(s)
15 immedsar TEXT        1 characters
16 winddir INTEGER      1 value(s)

Column definitions
# Name     Type       Length          Key
17 windspd INTEGER      1 value(s)
18 visdist INTEGER      1 value(s)
19 clouds  INTEGER      1 value(s)
20 ceiling INTEGER      1 value(s)
21 seas    INTEGER      1 value(s)
22 swells  INTEGER      1 value(s)
```

```
 Table: vslintel
 Read Password: NO
 Modify Password: NO

 Column definitions
 # Name     Type      Length         Key
 1 intvname TEXT       17 characters yes
 2 intdate  DATE        1 value(s)
 3 epiccode TEXT        1 characters
 4 stolen   TEXT        1 characters
 5 vremark1 TEXT       70 characters
 6 vremark2 TEXT       70 characters
 7 vremark3 TEXT       70 characters


 Table: vessel
 Read Password: NO
 Modify Password: NO

 Column definitions
 # Name     Type      Length         Key
 1 vslno    TEXT        8 characters
 2 vslname  TEXT       17 characters yes
 3 vslhin   TEXT       12 characters yes
 4 vslmake  TEXT       10 characters
 5 vslmodel TEXT       10 characters
 6 vslyear  TEXT        4 characters
 7 vsltons  INTEGER     1 value(s)
 8 vsllenft INTEGER     1 value(s)
 9 vsllenin INTEGER     1 value(s)
10 vslhp    TEXT        3 characters
11 vsluse   TEXT        2 characters
12 vsltype  TEXT        2 characters
13 vslprop  TEXT        1 characters
14 vslhull  TEXT        1 characters
15 vsleng   TEXT        1 characters
16 vslfuel  TEXT        1.characters

 Column definitions
 # Name     Type      Length         Key
17 vslconst TEXT        1 characters
18 cmedecal TEXT        1 characters
19 cmeyear  TEXT        4 characters
20 vsloffno TEXT       10 characters
21 homeport TEXT       25 characters
22 vslcall  TEXT       12 characters
23 vslvalue DOLLAR      1 value(s)
24 vslsuper TEXT       12 characters
25 vslmasts TEXT        1 characters
26 sailno   TEXT        8 characters
27 hlcolor  TEXT        3 characters
28 trcolor  TEXT        3 characters
29 slcolor  TEXT        3 characters
30 cbcolor  TEXT        3 characters
31 dkcolor  TEXT        3 characters
32 vslmarks TEXT       40 characters
```

```
Table: boarding
Read Password: NO
Modify Password: NO

Column definitions
 # Name      Type       Length          Key
 1 bodate    DATE        1 value(s)
 2 botime    TIME        1 value(s)
 3 bono      TEXT        7 characters  yes
 4 vslpob    INTEGER     1 value(s)
 5 adultpfd  INTEGER     1 value(s)
 6 childpfd  INTEGER     1 value(s)
 7 owstatus  TEXT        1 characters
 8 obsbody   TEXT       15 characters
 9 obscount  TEXT       15 characters
10 obscity   TEXT       13 characters
11 obsstate  TEXT        2 characters
12 obslat    REAL        1 value(s)
13 obslong   REAL        1 value(s)
14 viol54    TEXT        1 characters
15 viol55    TEXT        1 characters
16 viol56    TEXT        1 characters

Column definitions
 # Name      Type       Length          Key
17 viol57    TEXT        1 characters
18 viol58    TEXT        1 characters
19 viol59    TEXT        1 characters
20 viol60    TEXT        1 characters
21 viol61    TEXT        1 characters
22 viol62    TEXT        1 characters
23 viol63    TEXT        1 characters
24 viol64    TEXT        1 characters
25 viol65    TEXT        1 characters
26 viol66    TEXT        1 characters
27 viol67    TEXT        1 characters
28 viol68    TEXT        1 characters
29 uns69     TEXT        1 characters
30 uns70     TEXT        1 characters
31 uns71     TEXT        1 characters
32 uns72     TEXT        1 characters


Column definitions
 # Name      Type       Length          Key
33 uns73     TEXT        1 characters
34 uns74     TEXT        1 characters
35 uns75     TEXT        1 characters
36 uns76     TEXT        1 characters
37 uns77     TEXT        1 characters
38 uns78     TEXT        1 characters
39 uns79     TEXT        1 characters
40 borem1    TEXT       70 characters
41 borem2    TEXT       70 characters
42 borem3    TEXT       70 characters
43 boname    TEXT       25 characters
44 borate    TEXT        5 characters
45 bounit    TEXT       25 characters
46 bodist    TEXT        2 characters
47 boopfac   TEXT        5 characters
48 boday     TEXT        3 characters

Column definitions
 # Name      Type       Length          Key
49 bsunrise  TIME        1 value(s)
50 bsunset   TIME        1 value(s)
51 bovhin    TEXT       12 characters  yes
52 weapons   INTEGER     1 value(s)
```

```
Table: forms
Read Password: NO
Modify Password: YES

Column definitions
# Name     Type     Length         Key
1 fname    TEXT       8 characters yes
2 fdata    TEXT      80 characters


Table: reports
Read Password: NO
Modify Password: YES

Column definitions
# Name     Type     Length         Key
1 rname    TEXT       8 characters yes
2 rdata    TEXT     132 characters

Table: states
Read Password: NO
Modify Password: YES

Column definitions
# Name     Type     Length         Key
1 abbrev   TEXT       2 characters yes


Table: colors
Read Password: NO
Modify Password: YES

Column definitions
# Name     Type     Length         Key
1 shade    TEXT       3 characters yes
2 clrdesc  TEXT      20 characters


Table: days
Read Password: NO
Modify Password: YES

Column definitions
# Name     Type     Length         Key
1 dayabbr  TEXT       3 characters yes


Table: constr
Read Password: NO
Modify Password: YES

Column definitions
# Name     Type     Length         Key
1 constype TEXT       1 characters yes
2 consdesc TEXT      10 characters

Table: engcomp
Read Password: NO
Modify Password: YES

Column definitions
# Name     Type     Length         Key
1 ecomptyp TEXT       1 characters yes
2 ecompdes TEXT      10 characters
```

```
Table: fuelcomp
Read Password: NO
Modify Password: YES

Column definitions
# Name      Type      Length          Key
1 fcomptyp  TEXT       1 characters   yes
2 fcompdes  TEXT      10 characters


Table: hullmat
Read Password: NO
Modify Password: YES

Column definitions
# Name      Type      Length          Key
1 hmattype  TEXT       1 characters   yes
2 hmatdesc  TEXT      15 characters


Table: boattype
Read Password: NO
Modify Password: YES

Column definitions
# Name      Type      Length          Key
1 vtype     TEXT       2 characters   yes
2 vtypedes  TEXT      15 characters

Table: boatuse
Read Password: NO
Modify Password: YES

Column definitions
# Name      Type      Length          Key
1 vuse      TEXT       2 characters   yes
2 vusedesc  TEXT      15 characters


Table: boatprop
Read Password: NO
Modify Password: YES

Column definitions
# Name      Type      Length          Key
1 proptype  TEXT       1 characters   yes
2 propdesc  TEXT      15 characters


Table: status
Read Password: NO
Modify Password: YES

Column definitions
# Name      Type      Length          Key
1 ostattyp  TEXT       1 characters   yes
2 ostatdes  TEXT      30 characters
```

```
Table: courses
Read Password: NO
Modify Password: YES

Column definitions
# Name     Type      Length          Key
1 coursetp TEXT       1 characters   yes
2 coursede TEXT      15 characters

Table: majarea
Read Password: NO
Modify Password: YES

Column definitions
# Name     Type      Length          Key
1 majgrid  TEXT       2 characters   yes
2 majnorth REAL       1 value(s)
3 majsouth REAL       1 value(s)
4 majeast  REAL       1 value(s)
5 majwest  REAL       1 value(s)


Table: nations
Read Password: NO
Modify Password: YES

Column definitions
# Name     Type      Length          Key
1 symbol   TEXT       2 characters   yes
2 registry TEXT      30 characters


Table: minarea
Read Password: NO
Modify Password: YES

Column definitions
# Name     Type      Length          Key
1 mingrid  TEXT       2 characters   yes
2 minnorth REAL       1 value(s)
3 minsouth REAL       1 value(s)
4 mineast  REAL       1 value(s)
5 minwest  REAL       1 value(s)


Table: epic
Read Password: NO
Modify Password: YES

Column definitions
# Name     Type      Length          Key
1 epicno   TEXT       1 characters   yes
2 epicdsc1 TEXT      70 characters
3 epicdsc2 TEXT      70 characters
4 epicdsc3 TEXT      70 characters
5 epicdsc4 TEXT      70 characters
```

```
Table: rules
Read Password: NO
Modify Password: YES

Column definitions
# Name       Type      Length           Key
1 numrule    INTEGER    1 value(s)
2 and/or     TEXT       4 characters
3 colname1   TEXT       8 characters
4 table1     TEXT       8 characters
5 boolean    TEXT       4 characters
6 colname2   TEXT       8 characters
7 table2     TEXT       8 characters
8 rulvalue   TEXT      40 characters


Table: district
Read Password: NO
Modify Password: YES

Column definitions
# Name       Type      Length           Key
1 distno     TEXT       2 characters
2 distdesc   TEXT      35 characters


Table: sarunits
Read Password: NO
Modify Password: NO

Column definitions
# Name       Type      Length           Key
1 foldno     TEXT       7 characters    yes
2 unitname   TEXT      25 characters
3 uwtime     TIME       1 value( s)
4 uwdate     DATE       1 value( s)
5 oseta      TIME       1 value( s)
6 etadate    DATE       1 value( s)
7 ostime     TIME       1 value( s)
8 osdate     DATE       1 value( s)
9 alngtime   TIME       1 value( s)
10 alngdate  DATE       1 value( s)
11 towtime   TIME       1 value( s)
12 towdate   DATE       1 value( s)
13 dvmoored  TIME       1 value( s)
14 dvmdate   DATE       1 value( s)
15 sutmoor   TIME       1 value( s)
16 sutdate   DATE       1 value( s)

Table: viols
Read Password: NO
Modify Password: YES

Column definitions
# Name       Type      Length           Key
1 violno     TEXT       2 characters
2 violdesc   TEXT      40 characters
```

```
Table: opown
Read Password: NO
Modify Password: NO

Column definitions
# Name     Type     Length         Key
1 ownno    INTEGER    1 value(s)
2 opno     INTEGER    1 value(s)


Table: vardict
Read Password: NO
Modify Password: YES

Column definitions
# Name     Type     Length         Key
1 varname  TEXT       8 characters yes
2 vartype  TEXT       8 characters
3 varuse   TEXT      12 characters yes
4 vardesc  TEXT      25 characters
```

Report Format and Associated Table

REPORT NAME	*ASSOCIATED TABLE*
BOARDING	BOSUM
BOMAJOR	POSITFRQ
BOMINOR	POSITFRQ
BOQPRN	BOHIS
CMERPT	TEMP
DATADICT	DATADICT
DAYNIGHT	TEMP
EPICRPT	EPICTMP
EPICSUM	EPICSUM
IMMEDSAR	TEMP
INTELSUM	INTELSUM
OPERATOR	OPSUM2
OPERVSLS	OPERVSLS
OPQPRN	TEMP
OWNERS	OWNSUM2
OWNVSLS	OWNVSLS
OWQPRN	TEMP
PERINTEL	PERSUM
PERQPRN	TEMP
PINTQPRN	PERINT
SAR	SARSUM
SARDATES	SARSTAT
SARDAYS	TEMP
SARMAJOR	POSITFRQ
SARMINOR	POSITFRQ
SARQPRN	SARHIS
SARSMBT	TEMP
SARTIMES	TEMP
SMBOATRP	TEMP
STOLESUM	STOLESUM
UNSFREQ	TEMP
VARDICT	VARDICT
VCONSTMO	SUMMARY3
VENGMO	SUMMARY4
VESSEL	VSLSUM
VFUELMO	SUMMARY5
VHULLMO	SUMMARY6
VINTQPRN	VSLINT
VIOLFREQ	TEMP
VLENFTMO	TEMP
VPROPMO	SUMMARY7
VSLOPER	OPERVSLS
VSLOWNER	OWNVSLS
VSLQPRN	TEMP
VTYPEMO	SUMMARY1
VUSEMO	SUMMARY2
VYEARMO	TEMP

Database Attributes Sorted by Attribute Name

AttName	RelName	Type	Len	Attribute Description
abbrev	states	text	2	state abbreviation
adultpfd	boarding bosum	integer		number of adult pfd's on board
alngdate	sartime3 sartime4 sarunits	date		date alongside distressed vessel
alngtime	sartime3 sartime4 sarunits	time		alongside time of sar unit
and/or	rules	text	4	and/or conjunction in editting rules
attdesc	datadict	text	40	attribute description
attname	datadict	text	8	data dictionary attribute name
bodate	boarding bohis bosum violsum	date		date of boarding
boday	boarding bosum	text	3	boarding day of week
bodist	boarding bosum	text	2	boarding offcier district
boname	boarding bohis bosum	text	25	boarding officer name
bono	boarding bohis bosum bovslsum frqchart violsum vslmo	text	7	boarding report number
boolean	rules	text	4	boolean conjunction used in edit rule
boopfac	boarding bosum	text	5	boarding unit operating facility code
borate	boarding bohis bosum	text	5	boarding officer rank/rate
borem1	boarding bohis bosum	text	70	boarding officer remarks
borem2	boarding bohis bosum	text	70	boarding officer remarks
borem3	boarding bohis bosum	text	70	boarding officer remarks
botime	boarding bohis bosum bosum	time		time of boarding
bounit	boarding bosum	text	25	boarding officer unit
bovhin	boarding bohis bosum bovslsum violsum	text	12	vessel hull identification number
bsunrise	boarding bosum	time		time of sunrise
bsunset	boarding bosum	time		time of sunset

AttName	RelName	Type	Len	Attribute Description
cbcolor	opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	text	3	vessel cabin color
ceiling	sar sarsum	integer		on scene ceiling in feet
childpfd	boarding bosum	integer		number of child pfd's on board
clouds	sar sarsum	integer		on scene cloud coverage in tenths
clrdesc	colors	text	20	color abbreviation description
cmedecal	bovslsum opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	text	1	y or n for cme decal displayed
cmeyear	bovslsum opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	text	4	year of cme decal displayed
colname1	rules	text	8	column name used in editting rule
colname2	rules	text	8	column used in editting rule
consdesc	constr	text	10	vessel construction code description
constdes	summary3	text	10	vessel construction code description
constype	constr	text	1	vessel construction code
counter	summary1	integer		vessel type counter
coursede	courses	text	15	course description
coursetp	courses	text	1	course type code
dayabbr	days	text	3	day abbreviation
descprop	summary6	text	15	vessel propulsion description
distdesc	district	text	35	district number description
distno	district	text	2	district number code
dkcolor	opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	text	3	vessel deck color
dvmdate	sartime5 sarunits	date		date distressed vessel moored
dvmoored	sartime5 sarunits	time		time distressed vessel moored
ecnumber	epictmp	integer		count of total epic codes
ecompdes	engcomp	text	10	vessel engine compartment description
ecomptyp	engcomp	text	1	vessel engine compartment code
ecpct	epictmp	real		individual epic code percentage
engdesc	summary4	text	10	vessel engine compartment description

AttName	RelName	Type	Len	Attribute Description
epiccode	epicsum epictmp1 vslint vslintel	text	1	epic code
epicdsc1	epic epictmp epictmp2	text	70	description of epic code
epicdsc2	epic epictmp epictmp2	text	70	epic code description
epicdsc3	epic epictmp epictmp2	text	70	epic code description
epicdsc4	epic epictmp epictmp2	text	70	epic code description
epicno	epic epictmp epictmp2	text	1	epic code number
etadate	sarunits	date		date of estimated arrival
fcompdes	fuelcomp	text	10	vessel fuel compartment description
fcomptyp	fuelcomp	text	1	vessel fuel compartment code
fdata	forms	text	80	forms data
fname	forms	text	8	form name
folderno	sar sarhis sarsum sarsum1	text	7	sar folder number
foldno	sartime1 sartime2 sartime3 sartime4 sartime5 sarunits	text	7	sar folder number
freqcnt	positfrq positfrq	integer		major grid counter
fueldesc	summary5	text	10	vessel fuel compartment description
gridpct	positfrq positfrq	real		major grid percentage
hlcolor	opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	text	3	vessel hull color
hmatdesc	hullmat	text	15	vessel hull material description
hmattype	hullmat	text	1	vessel hull material code
homeport	opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	text	25	vessel homeport
hulldesc	summary6	text	15	vessel hull material description
idoper	opvsl opvsls1 opvsls2 pvsljoin vslsopr	integer		operator identification number (system)

AttName	RelName	Type	Len	Attribute Description
idowner	ownvsls1 ownvsls2 owvsl vslsown wvsljoin	integer		owner identification number (system)
immedsar	sar sarhis sarsum sarsum	text	1	y or n vessel in immediate danger
intdate	epicsum epictmp1 stolesum vslint vslintel	date		date intelligence received/entered
intdob	perint perintel persum	date		date of birth of person
intfname	perint perintel persum	text	15	first name of person
intlname	perint perintel persum	text	20	last name of person
intmi	perint perintel persum	text	1	middle initial of person
intvdate	intelsum	date		date intelligence received/entered
intvname	epicsum intelsum stolesum vslint vslintel	text	17	vessel name
length	datadict	text	3	data dictionary attribute length
majeast	majarea positfrq	real		eastern bound of major grid
majgrid	majarea positfrq	text	2	designation of major grid area
majnorth	majarea positfrq	real		northern bound of major grid
majsouth	majarea positfrq	real		southern bound of major grid
majwest	majarea positfrq	real		western bound of major area
mineast	minarea positfrq	real		eastern bound of minor grid
mingrid	minarea positfrq	text	2	designation of minor grid area
minnorth	minarea positfrq	real		northern bound of minor grid
minsouth	minarea positfrq	real		southern bound of minor grid
minwest	minarea positfrq	real		western bound of minor grid
mucn	sar sarhis sarsum	text	8	multi-unit case number
numconst	summary3	integer		vessel construction counter
numeng	summary4	integer		vessel engine compartment counter
numfuel	summary5	integer		vessel fuel compartment counter
numhull	summary6	integer		vessel hull material counter
numprop	summary7	integer		vessel propulsion counter
numrule	rules	integer		number of input editting rule
numuse	summary2	integer		vessel use counter

AttName	RelName	Type	Len	Attribute Description
obsbody	boarding bosum	text	15	body of water that boarding occurred
obscity	boarding bosum	text	13	city that boarding occurred
obscount	boarding bosum	text	15	county that boarding occurred
obslat	boarding bohis bosum	real	2.3	latitude where boarding occurred
obslong	boarding bohis bosum	real	3.3	longitude where boarding occurred
obsstate	boarding bosum	text	2	coded state abbrev that boarding occurs
offshore	sar sarhis sarsum	integer		distance in nm offshore
opaddr	operator opervsls opsum1 opsum2 opvsls1 opvsls2	text	25	operator street address
opcity	operator opervsls opsum1 opsum2 opvsls1 opvsls2	text	18	operator city address
opcourse	operator opsum1 opsum2 opvsls1 opvsls2	text	1	coded operator courses
opdob	operator opervsls opsum1 opsum2 opvsls1 opvsls2	date		operator date of birth
opfname	operator opervsls opsum1 opsum2 opvsls1 opvsls2	text	15	operator first name
opid	operator opsum1 opsum2 opvsls1 opvsls2	integer		operator identification number (system)
oplname	operator opervsls opsum1 opsum2 opvsls1 opvsls2	text	20	operator last name
opmi	operator opervsls opsum1 opsum2 opvsls1 opvsls2	text	1	operator middle initial
opno	opown	integer		operator identification number (system)

AttName	RelName	Type	Len	Attribute Description
opstate	operator	text	2	coded operator state abbrev.
	opervsls			
	opsum1			
	opsum2			
	opvsls1			
	opvsls2			
optelno	operator	text	12	operator telephone number
	opervsls			
	opsum1			
	opsum2			
	opvsls1			
	opvsls2			
opzip	operator	text	9	operator zip code address
	opervsls			
	opsum1			
	opsum2			
	opvsls1			
	opvsls2			
osdate	sartime1	date		date on-scene
	sartime4			
	sarunits			
oseta	sarunits	time		on scene eta of sar unit
ostatdes	status	text	30	owner status description
ostattyp	status	text	1	owner status code
ostime	sartime1	time		on scene time of sar unit
	sartime4			
	sarunits			
ownaddr	owners	text	25	owner street address
	ownsum1			
	ownsum2			
	ownvsls			
	ownvsls1			
	ownvsls2			
owncity	owners	text	18	owner city address
	ownsum1			
	ownsum2			
	ownvsls			
	ownvsls1			
	ownvsls2			
owndob	owners	date		owner date of birth
	ownsum1			
	ownsum2			
	ownvsls			
	ownvsls1			
	ownvsls2			
ownfname	owners	text	15	owner first name
	ownsum1			
	ownsum2			
	ownvsls			
	ownvsls1			
	ownvsls2			
ownid	owners	integer		owner identification number (system)
	ownsum1			
	ownsum2			
	ownvsls1			
	ownvsls2			
ownlname	owners	text	20	owner last name
	ownsum1			
	ownsum2			
	ownvsls			
	ownvsls1			
	ownvsls2			

AttName	RelName	Type	Len	Attribute Description
ownmi	owners ownsum1 ownsum2 ownvsls ownvsls1 ownvsls2	text	1	owner middle initial
ownno	opown	integer		owner identification number (system)
ownstate	owners ownsum1 ownsum2 ownvsls ownvsls1 ownvsls2	text	2	coded owner state abbreviation address
owntelno	owners ownsum1 ownsum2 ownvsls ownvsls1 ownvsls2	text	12	owner telephone number
ownzip	owners ownsum1 ownsum2 ownvsls ownvsls1 ownvsls2	text	9	owner zip code address
owstatus	boarding bosum	text	1	coded owner status at time of boarding
pctincr	sarstat	real		percentage increase in sar cases
pintdate	perint perintel persum	date		date intelligence received/entered
pintrem1	perint perintel persum	text	70	intelligence remarks on person
pintrem2	perint perintel persum	text	70	intelligence remarks on person
pintrem3	perint perintel persum	text	70	intelligence remarks on person
propdesc	boatprop	text	15	vessel propulsion description
proptype	boatprop	text	1	vessel propulsion code
rdata	reports	text	132	report data
registry	nations	text	30	nation description
relname	datadict	text	8	data dictionary relation name
rname	reports	text	8	report name
rulvalue	rules	text	40	edit error message
sailno	opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	text	8	sail number of vessel
sardate	sar sarhis sarsum sarsum1	date		date of sar
sarday	sar sarsum sarsum1	text	3	sar day of week abbreviated
sarlat	sar sarhis sarsum	real	2.3	latitude position of sar

AttName	RelName	Type	Len	Attribute Description
sarlong	sar sarhis sarsum	real	3.3	longitude position of sar
sarnod	sar sarhis sarsum	text	25	sar nature of distress
sarnum	sarstat	integer		number of sar cases
sarpob	sar sarhis sarsum	integer		number of personnel onboard
sartime	sar sarhis sarsum	time		time sar report received
sarvhin	sar sarhis sarsum	text	12	vessel hull identification number
saryear	sarstat	text	4	year of sar case
seas	sar sarsum	integer		on scene sea height in feet
shade	colors	text	3	color abbreviation
slcolor	opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	text	3	vessel sail color
source	datadict	text	15	source document
ssunrise	sar sarsum	time		time of sunrise
ssunset	sar sarsum	time		time of sunset
stolen	epicsum stolesum vslint vslintel	text	1	y or n for vessel stolen
sutdate	sartime2 sarunits	date		date sar unit moored/ returned
sutmoor	sartime2 sarunits	time		time sar unit moored
swells	sar sarsum	integer		on scene swell height in feet
symbol	nations	text	2	nation abbreviation
table1	rules	text	8	table used in editting rule
table2	rules	text	8	table used in editting rule
tmp	temp	text	2	attribute/relation for report printing
towdate	sartime5 sarunits	date		date distressed vessel in tow
towtime	sartime5 sarunits	time		time distressed vessel in tow
trcolor	opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	text	3	vessel trim color
type	datadict	text	7	data dictionary attribute type
typedesc	summary1	text	15	vessel type description
ucn	sar sarhis sarsum	text	8	unit case number

AttName	RelName	Type	Len	Attribute Description
unitname	sarunits	text	25	sar unit name
uns69	boarding bohis bosum frqchart violsum	text	1	'X' if vessel terminated
uns70	boarding bohis bosum frqchart violsum	text	1	'X' if condition corrected on the spot
uns71	boarding bohis bosum frqchart	text	1	'X' if vessel overloaded
uns72	boarding bohis bosum frqchart violsum	text	1	'X' if fuel in bilges
uns73	boarding bohis bosum frqchart violsum	text	1	'X' if fuel leak
uns74	boarding bohis bosum frqchart violsum	text	1	'X' if manifestly unsafe voyage
uns75	boarding bohis bosum frqchart violsum	text	1	'X' if hazardous bars (13th district)
uns76	boarding bohis bosum frqchart violsum	text	1	'X' if operator alcohol condition
uns77	boarding bohis bosum frqchart violsum	text	1	'X' if other unsafe condition
uns78	boarding bohis bosum frqchart violsum	text	1	'X' if warning issued
uns79	boarding bohis bosum frqchart violsum	text	1	'X' if no violations
usedesc	summary2	text	15	vessel use description
uwdate	sartime1 sartime2 sartime3 sarunits	date		sar unit underway date
uwtime	sartime1 sartime2 sartime3 sarunits	time		underway time of sar unit

AttName	RelName	Type	Len	Attribute Description
vardesc	vardict	text	25	attribute represented or how used
varname	vardict	text	8	name of variable
vartype	vardict	text	8	type of variable
varusel	vardict	text	12	where the variable is used
vhinoper	opvsl opvsls1 opvsls2 pvsljoin vslsopr	text	12	vessel hull identification
vhinown	ownvsls1 ownvsls2 owvsl vslsown wvsljoin	text	12	vessel hull identification number
viol54	boarding bohis bosum frqchart violsum	text	1	'X' if numbering violation
viol55	boarding bohis bosum frqchart violsum	text	1	'X' if certificate violation
viol56	boarding bohis bosum frqchart violsum	text	1	'X' if pfd violation
viol57	boarding bohis bosum frqchart violsum	text	1	'X' if sound producing device violation
viol58	boarding bohis bosum frqchart violsum	text	1	'X' if bell violation
viol59	boarding bohis bosum frqchart violsum	text	1	'X' if fire extinguisher violation
viol60	boarding bohis bosum frqchart violsum	text	1	'X' if flame control violation
viol61	boarding bohis bosum frqchart violsum	text	1	'X' if ventilation violation
viol62	boarding bohis bosum frqchart violsum	text	1	'X' if navigation/anchor lts. violation
viol63	boarding bohis bosum frqchart violsum	text	1	'X' if negligent operation violation

AttName	RelName	Type	Len	Attribute Description
viol64	boarding bohis bosum frqchart violsum	text	1	'X' if failure to terminate violation
viol65	boarding bohis bosum frqchart violsum	text	1	'X' if pollution placard not posted
viol66	boarding bohis bosum frqchart violsum	text	1	'X' if marine sanitation device violtn.
viol67	boarding bohis bosum frqchart violsum	text	1	'X' if visual distress signal violation
viol68	boarding bohis bosum frqchart violsum	text	1	'X' if document/official no. violation
violdesc	viols	text	40	violation description
violno	viols	text	2	violation number
visdist	sar sarsum	integer		visibility distance
vremark1	intelsum vslint vslintel	text	70	vessel intelligence remarks
vremark2	intelsum vslint vslintel	text	70	vessel intelligence remarks
vremark3	intelsum vslint vslintel	text	70	vessel intelligence remarks
vslcall	opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	text	12	vessel call sign
vslconst	opvsls2 ownvsls2 pvsljoin summary3 vessel violsum vslmo vslsum vslsum1 vslsum2 wvsljoin	text	1	coded vessel construction type

AttName	RelName	Type	Len	Attribute Description
vsleng	opvsls2 ownvsls2 pvsljoin summary4 vessel violsum vslmo vslsum vslsum1 vslsum2 wvsljoin	text	1	coded vessel engine compartment type
vslfuel	opvsls2 ownvsls2 pvsljoin summary5 vessel violsum vslmo vslsum vslsum1 vslsum2 wvsljoin	text	1	coded vessel fuel compartment type
vslhin	bovslsum opervsls opvsls2 ownvsls ownvsls2 pvsljoin vessel violsum vslsum vslsum1 vslsum2 wvsljoin	text	12	vessel hull identification number
vslhp	opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	text	3	vessel horsepower
vslhull	opvsls2 ownvsls2 pvsljoin summary6 vessel violsum vslmo vslsum vslsum1 vslsum2 wvsljoin	text	1	coded vessel hull material
vsllenft	opvsls2 ownvsls2 pvsljoin vessel violsum vslmo vslsum vslsum1 vslsum2 wvsljoin	integer		vessel length in feet

AttName	RelName	Type	Len	Attribute Description
vsllenin	opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	integer		vessel length inches
vslmake	opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	text	10	vessel make
vslmarks	opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	text	40	vessel unusual markings
vslmasts	opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	text	1	number of masts on vessel
vslmodel	opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	text	10	vessel model
vslname	opervsls opvsls2 ownvsls ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	text	17	vessel name
vslno	opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	text	8	vessel number
vsloffno	opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	text	10	vessel official number (main beam no.)

AttName	RelName	Type	Len	Attribute Description
vslpob	boarding bohis bosum	integer		number of personnel on board vessel
vslprop	opvsls2 ownvsls2 pvsljoin summary6 vessel violsum vslmo vslsum vslsum1 vslsum2 wvsljoin	text	1	coded vessel propulsion method
vslsuper	opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	text	12	vessel superstructure
vsltons	opvsls2 ownvsls2 pvsljoin vessel violsum vslsum vslsum1 vslsum2 wvsljoin	integer		vessel net tonnage
vsltype	opvsls2 ownvsls2 pvsljoin summary1 vessel violsum vslmo vslsum vslsum1 vslsum2 wvsljoin	text	2	coded vessel type
vsluse	opvsls2 ownvsls2 pvsljoin summary2 vessel violsum vslmo vslsum vslsum1 vslsum2 wvsljoin	text	2	coded vessel use
vslvalue	opvsls2 ownvsls2 pvsljoin vessel vslsum vslsum1 vslsum2 wvsljoin	dollar		vessel value

AttName	RelName	Type	Len	Attribute Description
vslyear	opvsls2	text	4	vessel model year
	ownvsls2			
	pvsljoin			
	vessel			
	violsum			
	vslmo			
	vslsum			
	vslsum1			
	vslsum2			
	wvsljoin			
vtype	boattype	text	2	vessel type code
vtypedes	boattype	text	15	vessel type description
vuse	boatuse	text	2	vessel use code
vusedesc	boatuse	text	15	vessel use description
weapons	boarding	integer		number of weapons on board
	bohis			
	bosum			
	violsum			
winddir	sar	integer		wind direction
	sarsum			
windspd	sar	integer		wind speed
	sarsum			

Database Attributes Sorted by Input Source

Source	AttName	RelName	Type	Len
4100	adultpfd	boarding	integer	
	bodate	boarding	date	
	boday	boarding	text	3
	bodist	boarding	text	2
	boname	boarding	text	25
	bono	boarding	text	7
	boopfac	boarding	text	5
	borate	boarding	text	5
	borem1	boarding	text	70
	borem2	boarding	text	70
	borem3	boarding	text	70
	botime	boarding	time	
	bounit	boarding	text	25
	bovhin	boarding	text	12
	bsunrise	boarding	time	
	bsunset	boarding	time	
	childpfd	boarding	integer	
	cmedecal	vessel	text	1
	cmeyear	vessel	text	4
	obsbody	boarding	text	15
	obscity	boarding	text	13
	obscount	boarding	text	15
	obslat	boarding	real	2.3
	obslong	boarding	real	3.3
	obsstate	boarding	text	2
	opaddr	operator	text	25
	opcity	operator	text	18
	opcourse	operator	text	1
	opdob	operator	date	
	opfname	operator	text	15
	oplname	operator	text	20
	opmi	operator	text	1
	opstate	operator	text	2
	optelno	operator	text	12
	opzip	operator	text	9
	ownaddr	owners	text	25
	owncity	owners	text	18
	owndob	owners	date	
	ownfname	owners	text	15
	ownlname	owners	text	20
	ownmi	owners	text	1
	ownstate	owners	text	2
	owntelno	owners	text	12
	ownzip	owners	text	9
	owstatus	boarding	text	1
	uns69	boarding	text	1
	uns70	boarding	text	1
	uns71	boarding	text	1
	uns72	boarding	text	1
	uns73	boarding	text	1
	uns74	boarding	text	1
	uns75	boarding	text	1
	uns76	boarding	text	1
	uns77	boarding	text	1
	uns78	boarding	text	1
	uns79	boarding	text	1
	vhinoper	opvsl	text	12
	vhinown	owvsl	text	12
	viol54	boarding	text	1
	viol55	boarding	text	1
	viol56	boarding	text	1
	viol57	boarding	text	1
	viol58	boarding	text	1
	viol59	boarding	text	1
	viol60	boarding	text	1

Source	AttName	RelName	Type	Len
4100	viol61	boarding	text	1
	viol62	boarding	text	1
	viol63	boarding	text	1
	viol64	boarding	text	1
	viol65	boarding	text	1
	viol66	boarding	text	1
	viol67	boarding	text	1
	viol68	boarding	text	1
	vslconst	vessel	text	1
	vsleng	vessel	text	1
	vslfuel	vessel	text	1
	vslhin	vessel	text	12
	vslhp	vessel	text	3
	vslhull	vessel	text	1
	vsllenft	vessel	integer	
	vsllenin	vessel	integer	
	vslmake	vessel	text	10
	vslmodel	vessel	text	10
	vslname	vessel	text	17
	vslno	vessel	text	8
	vslpob	boarding	integer	
	vslprop	vessel	text	1
	vsltons	vessel	integer	
	vsltype	vessel	text	2
	vsluse	vessel	text	2
	vslyear	vessel	text	4
	weapons	boarding	integer	
designer	abbrev	states	text	2
	and/or	rules	text	4
	attdesc	datadict	text	40
	attname	datadict	text	8
	boolean	rules	text	4
	clrdesc	colors	text	20
	colname1	rules	text	8
	colname2	rules	text	8
	consdesc	constr	text	10
	constype	constr	text	1
	coursede	courses	text	15
	coursetp	courses	text	1
	dayabbr	days	text	3
	distdesc	district	text	35
	distno	district	text	2
	ecompdes	engcomp	text	10
	ecomptyp	engcomp	text	1
	epicdsc1	epic	text	70
	epicdsc2	epic	text	70
	epicdsc3	epic	text	70
	epicdsc4	epic	text	70
	epicno	epic	text	1
	fcompdes	fuelcomp	text	10
	fcomptyp	fuelcomp	text	1
	fdata	forms	text	80
	fname	forms	text	8
	hmatdesc	hullmat	text	15
	hmattype	hullmat	text	1
	length	datadict	text	3
	majeast	majarea	real	
	majgrid	majarea	text	2
	majnorth	majarea	real	
	majsouth	majarea	real	
	majwest	majarea	real	
	mineast	minarea	real	
	mingrid	minarea	text	2
	minnorth	minarea	real	
	minsouth	minarea	real	

Source	AttName	RelName	Type	Len
designer	minwest	minarea	real	
	numrule	rules	integer	
	ostatdes	status	text	30
	ostattyp	status	text	1
	propdesc	boatprop	text	15
	proptype	boatprop	text	1
	rdata	reports	text	132
	registry	nations	text	30
	relname	datadict	text	8
	rname	reports	text	8
	rulvalue	rules	text	40
	shade	colors	text	3
	source	datadict	text	15
	symbol	nations	text	2
	table1	rules	text	8
	table2	rules	text	8
	type	datadict	text	7
	vardesc	vardict	text	25
	varname	vardict	text	8
	vartype	vardict	text	8
	varusel	vardict	text	12
	violdesc	viols	text	40
	violno	viols	text	2
	vtype	boattype	text	2
	vtypedes	boattype	text	15
	vuse	boatuse	text	2
	vusedesc	boatuse	text	15
numerous	epiccode	vslintel	text	1
	intdate	vslintel	date	
	intdob	perintel	date	
	intfname	perintel	text	15
	intlname	perintel	text	20
	intmi	perintel	text	1
	intvname	vslintel	text	17
	pintdate	perintel	date	
	pintrem1	perintel	text	70
	pintrem2	perintel	text	70
	pintrem3	perintel	text	70
	stolen	vslintel	text	1
	vremark1	vslintel	text	70
	vremark2	vslintel	text	70
	vremark3	vslintel	text	70
ood	ssunrise	sar	time	
	ssunset	sar	time	
sar fold	alngdate	sarunits	date	
	alngtime	sarunits	time	
	cbcolor	vessel	text	3
	ceiling	sar	integer	
	clouds	sar	integer	
	dkcolor	vessel	text	3
	dvmdate	sarunits	date	
	dvmoored	sarunits	time	
	etadate	sarunits	date	
	folderno	sar	text	7
	foldno	sarunits	text	7
	hlcolor	vessel	text	3
	homeport	vessel	text	25
	immedsar	sar	text	1
	mucn	sar	text	8
	offshore	sar	integer	
	osdate	sarunits	date	
	oseta	sarunits	time	
	ostime	sarunits	time	
	sailno	vessel	text	8
	sardate	sar	date	

Source	AttName	RelName	Type	Len
sar fold	sarday	sar	text	3
	sarlat	sar	real	2.3
	sarlong	sar	real	3.3
	sarnod	sar	text	25
	sarpob	sar	integer	
	sartime	sar	time	
	sarvhin	sar	text	12
	seas	sar	integer	
	slcolor	vessel	text	3
	sutdate	sarunits	date	
	sutmoor	sarunits	time	
	swells	sar	integer	
	towdate	sarunits	date	
	towtime	sarunits	time	
	trcolor	vessel	text	3
	ucn	sar	text	8
	unitname	sarunits	text	25
	uwdate	sarunits	date	
	uwtime	sarunits	time	
	visdist	sar	integer	
	vslcall	vessel	text	12
	vslmarks	vessel	text	40
	vslmasts	vessel	text	1
	vsloffno	vessel	text	10
	vslsuper	vessel	text	12
	vslvalue	vessel	dollar	
	winddir	sar	integer	
	windspd	sar	integer	
system	idoper	opvsl	integer	
	idowner	owvsl	integer	
	opid	operator	integer	
	opno	opown	integer	
	ownid	owners	integer	
	ownno	opown	integer	

Database Attributes Sorted by Where Used

Used In	AttName	RelName	Type	Len
boarding	adultpfd	bosum	integer	
	bodate	bosum	date	
	boday	bosum	text	3
	bodist	bosum	text	2
	boname	bosum	text	25
	bono	bosum	text	7
	boopfac	bosum	text	5
	borate	bosum	text	5
	borem1	bosum	text	70
	borem2	bosum	text	70
	borem3	bosum	text	70
	botime	bosum	text	
	bounit	bosum	text	25
	bovhin	bosum	text	12
	bsunrise	bosum	time	
	bsunset	bosum	time	
	childpfd	bosum	integer	
	obsbody	bosum	text	15
	obscity	bosum	text	13
	obscount	bosum	text	15
	obslat	bosum	real	2.3
	obslong	bosum	real	3.3
	obsstate	bosum	text	2
	owstatus	bosum	text	1
	uns69	bosum	text	1
	uns70	bosum	text	1
	uns72	bosum	text	1
	uns73	bosum	text	1
	uns74	bosum	text	1
	uns75	bosum	text	1
	uns76	bosum	text	1
	uns77	bosum	text	1
	uns78	bosum	text	1
	uns79	bosum	text	1
	viol54	bosum	text	1
	viol55	bosum	text	1
	viol56	bosum	text	1
	viol57	bosum	text	1
	viol58	bosum	text	1
	viol59	bosum	text	1
	viol60	bosum	text	1
	viol61	bosum	text	1
	viol62	bosum	text	1
	viol63	bosum	text	1
	viol64	bosum	text	1
	viol65	bosum	text	1
	viol66	bosum	text	1
	viol67	bosum	text	1
	viol68	bosum	text	1
	vslpob	bosum	integer	
	weapons	bosum	integer	
bomajor	bono	bosum	text	7
	freqcnt	positfrq	integer	
	gridpct	positfrq	real	
	majeast	positfrq	real	3.3
	majgrid	positfrq	text	2
	majnorth	positfrq	real	2.3
	majsouth	positfrq	real	2.3
	majwest	positfrq	real	3.3
	obslat	bosum	real	2.3
	obslong	bosum	real	3.3
bominor	bono	bosum	text	7
	freqcnt	positfrq	integer	
	gridpct	positfrq	real	
	mineast	positfrq	real	3.3

Used In	AttName	RelName	Type	Len
bominor	mingrid	positfrq	text	2
	minnorth	positfrq	real	2.3
	minsouth	positfrq	real	2.3
	minwest	positfrq	real	3.3
	obslat	bosum	real	2.3
	obslong	bosum	real	3.3
cmerpt	bono	bosum	text	7
	bono	bovslsum	text	7
	bovhin	bosum	text	12
	bovhin	bovslsum	text	12
	cmedecal	bovslsum	text	1
	cmedecal	vslsum	text	1
	cmeyear	bovslsum	text	4
	cmeyear	vslsum	text	4
	tmp	temp	text	2
	vslhin	bovslsum	text	12
	vslhin	vslsum	text	12
daynight	bodate	bosum	date	
	botime	bosum	time	
	bsunrise	bosum	time	
	bsunset	bosum	time	
	sardate	sarsum	date	
	sartime	sarsum	time	
	ssunrise	sarsum	time	
	ssunset	sarsum	time	
	tmp	temp	text	2
epicdump	ecnumber	epictmp	integer	
	ecpct	epictmp	real	
	epiccode	epicsum	text	1
	epiccode	epictmp1	text	1
	epicdsc1	epictmp	text	70
	epicdsc1	epictmp2	text	70
	epicdsc2	epictmp	text	70
	epicdsc2	epictmp2	text	70
	epicdsc3	epictmp	text	70
	epicdsc3	epictmp2	text	70
	epicdsc4	epictmp	text	70
	epicdsc4	epictmp2	text	70
	epicno	epictmp	text	1
	epicno	epictmp2	text	1
	intdate	epicsum	date	
	intdate	epictmp1	date	
	intdate	stolesum	date	
	intvdate	intelsum	date	
	intvname	epicsum	text	17
	intvname	intelsum	text	17
	intvname	stolesum	text	17
	stolen	epicsum	text	1
	stolen	stolesum	text	1
	vremark1	intelsum	text	70
	vremark2	intelsum	text	70
	vremark3	intelsum	text	70
immedsar	folderno	sarsum	text	7
	immedsar	sarsum	real	1
	sardate	sarsum	date	
	tmp	temp	text	2
intelrpt	bodate	bosum	date	
	bodate	violsum	date	
	bono	bosum	text	7
	bono	violsum	text	7
	bono	vslmo	text	7
	bovhin	bosum	text	12
	bovhin	violsum	text	12
	constdes	summary3	text	10
	counter	summary1	integer	

Used In	AttName	RelName	Type	Len
intelrpt	descprop	summary6	text	15
	engdesc	summary4	text	10
	fueldesc	summary5	text	10
	hulldesc	summary6	text	15
	numconst	summary3	integer	
	numeng	summary4	integer	
	numfuel	summary5	integer	
	numhull	summary6	integer	
	numprop	summary7	integer	
	numuse	summary2	integer	
	tmp	temp	text	2
	typedesc	summary1	text	15
	uns69	bosum	text	1
	uns69	violsum	text	1
	uns70	bosum	text	1
	uns70	violsum	text	1
	uns71	bosum	text	1
	uns72	bosum	text	1
	uns72	violsum	text	1
	uns73	bosum	text	1
	uns73	violsum	text	1
	uns74	bosum	text	1
	uns74	violsum	text	1
	uns75	bosum	text	1
	uns75	violsum	text	1
	uns76	bosum	text	1
	uns76	violsum	text	1
	uns77	bosum	text	1
	uns77	violsum	text	1
	uns78	bosum	text	1
	uns78	violsum	text	1
	uns79	bosum	text	1
	uns79	violsum	text	1
	usedesc	summary2	text	15
	viol54	bosum	text	1
	viol54	violsum	text	1
	viol55	bosum	text	1
	viol55	violsum	text	1
	viol56	bosum	text	1
	viol56	violsum	text	1
	viol57	bosum	text	1
	viol57	violsum	text	1
	viol58	bosum	text	1
	viol58	violsum	text	1
	viol59	bosum	text	1
	viol59	violsum	text	1
	viol60	bosum	text	1
	viol60	violsum	text	1
	viol61	bosum	text	1
	viol61	violsum	text	1
	viol62	bosum	text	1
	viol62	violsum	text	1
	viol63	bosum	text	1
	viol63	violsum	text	1
	viol64	bosum	text	1
	viol64	violsum	text	1
	viol65	bosum	text	1
	viol65	violsum	text	1
	viol66	bosum	text	1
	viol66	violsum	text	1
	viol67	bosum	text	1
	viol67	violsum	text	1
	viol68	bosum	text	1
	viol68	violsum	text	1
	vslconst	summary3	text	2

Used In	AttName	RelName	Type	Len
intelrpt	vslconst	violsum	text	1
	vslconst	vslmo	text	1
	vslconst	vslsum	text	1
	vsleng	summary4	text	1
	vsleng	violsum	text	1
	vsleng	vslmo	text	1
	vsleng	vslsum	text	1
	vslfuel	summary5	text	1
	vslfuel	violsum	text	1
	vslfuel	vslmo	text	1
	vslfuel	vslsum	text	1
	vslhin	violsum	text	12
	vslhin	vslsum	text	12
	vslhull	summary6	text	1
	vslhull	violsum	text	1
	vslhull	vslmo	text	1
	vslhull	vslsum	text	1
	vsllenft	violsum	integer	
	vsllenft	vslmo	integer	
	vsllenft	vslsum	integer	
	vslprop	summary6	text	1
	vslprop	violsum	text	1
	vslprop	vslmo	text	1
	vslprop	vslsum	text	1
	vsltons	violsum	integer	
	vsltons	vslsum	integer	
	vsltype	summary1	text	2
	vsltype	violsum	text	2
	vsltype	vslmo	text	2
	vsltype	vslsum	text	2
	vsluse	summary2	text	2
	vsluse	violsum	text	2
	vsluse	vslmo	text	2
	vsluse	vslsum	text	2
	vslyear	violsum	text	4
	vslyear	vslmo	text	4
	vslyear	vslsum	text	4
	weapons	bosum	integer	
	weapons	violsum	integer	
operator	opaddr	opsum1	text	25
	opaddr	opsum2	text	25
	opcity	opsum1	text	18
	opcity	opsum2	text	18
	opcourse	opsum1	text	1
	opcourse	opsum2	text	1
	opdob	opsum1	date	
	opdob	opsum2	date	
	opfname	opsum1	text	15
	opfname	opsum2	text	15
	opid	opsum1	integer	
	opid	opsum2	integer	
	oplname	opsum1	text	20
	oplname	opsum2	text	20
	opmi	opsum1	text	1
	opmi	opsum2	text	1
	opstate	opsum1	text	2
	opstate	opsum2	text	2
	optelno	opsum1	text	12
	optelno	opsum2	text	12
	opzip	opsum1	text	9
	opzip	opsum2	text	9

Used In	AttName	RelName	Type	Len
opervsls	cbcolor	opvsls2	text	3
	cmedecal	opvsls2	text	1
	cmeyear	opvsls2	text	4
	dkcolor	opvsls2	text	3
	hlcolor	opvsls2	text	3
	homeport	opvsls2	text	25
	idoper	opvsls1	integer	
	idoper	opvsls2	integer	
	opaddr	opervsls	text	25
	opaddr	opvsls1	text	25
	opaddr	opvsls2	text	25
	opcity	opervsls	text	18
	opcity	opvsls1	text	18
	opcity	opvsls2	text	18
	opcourse	opvsls1	text	1
	opcourse	opvsls2	text	1
	opdob	opervsls	date	
	opdob	opvsls1	date	
	opdob	opvsls2	date	
	opfname	opervsls	text	15
	opfname	opvsls1	text	15
	opfname	opvsls2	text	15
	opid	opvsls1	integer	
	opid	opvsls2	integer	
	oplname	opervsls	text	20
	oplname	opvsls1	text	20
	oplname	opvsls2	text	20
	opmi	opervsls	text	1
	opmi	opvsls1	text	1
	opmi	opvsls2	text	1
	opstate	opervsls	text	2
	opstate	opvsls1	text	2
	opstate	opvsls2	text	2
	optelno	opervsls	text	12
	optelno	opvsls1	text	12
	optelno	opvsls2	text	12
	opzip	opervsls	text	9
	opzip	opvsls1	text	9
	opzip	opvsls2	text	9
	sailno	opvsls2	text	8
	slcolor	opvsls2	text	3
	trcolor	opvsls2	text	3
	vhinoper	opvsls1	text	12
	vhinoper	opvsls2	text	12
	vslcall	opvsls2	text	12
	vslconst	opvsls2	text	1
	vsleng	opvsls2	text	1
	vslfuel	opvsls2	text	1
	vslhin	opervsls	text	12
	vslhin	opvsls2	text	12
	vslhp	opvsls2	text	3
	vslhull	opvsls2	text	1
	vsllenft	opvsls2	integer	
	vsllenin	opvsls2	integer	
	vslmake	opvsls2	text	10
	vslmarks	opvsls2	text	40
	vslmasts	opvsls2	text	1
	vslmodel	opvsls2	text	10
	vslname	opervsls	text	17
	vslname	opvsls2	text	17
	vslno	opvsls2	text	8
	vsloffno	opvsls2	text	10
	vslprop	opvsls2	text	1
	vslsuper	opvsls2	text	12
	vsltons	opvsls2	integer	

Used In	AttName	RelName	Type	Len
opervsls	vsltype	opvsls2	text	2
	vsluse	opvsls2	text	2
	vslvalue	opvsls2	dollar	
	vslyear	opvsls2	text	4
owners	ownaddr	ownsum1	text	25
	ownaddr	ownsum2	text	25
	owncity	ownsum1	text	18
	owncity	ownsum2	text	18
	owndob	ownsum1	date	
	owndob	ownsum2	date	
	ownfname	ownsum1	text	15
	ownfname	ownsum2	text	15
	ownid	ownsum1	integer	
	ownid	ownsum2	integer	
	ownlname	ownsum1	text	20
	ownlname	ownsum2	text	20
	ownmi	ownsum1	text	1
	ownmi	ownsum2	text	1
	ownstate	ownsum1	text	2
	ownstate	ownsum2	text	2
	owntelno	ownsum1	text	12
	owntelno	ownsum2	text	12
	ownzip	ownsum1	text	9
	ownzip	ownsum2	text	9
ownervsl	cbcolor	ownvsls2	text	3
	cmedecal	ownvsls2	text	1
	cmeyear	ownvsls2	text	4
	dkcolor	ownvsls2	text	3
	hlcolor	ownvsls2	text	3
	homeport	ownvsls2	text	25
	idowner	ownvsls1	integer	
	idowner	ownvsls2	integer	
	ownaddr	ownvsls	text	25
	ownaddr	ownvsls1	text	25
	ownaddr	ownvsls2	text	25
	owncity	ownvsls	text	18
	owncity	ownvsls1	text	18
	owncity	ownvsls2	text	18
	owndob	ownvsls	date	
	owndob	ownvsls1	date	
	owndob	ownvsls2	date	
	ownfname	ownvsls	text	15
	ownfname	ownvsls1	text	15
	ownfname	ownvsls2	text	15
	ownid	ownvsls1	integer	
	ownid	ownvsls2	integer	
	ownlname	ownvsls	text	20
	ownlname	ownvsls1	text	20
	ownlname	ownvsls2	text	20
	ownmi	ownvsls	text	1
	ownmi	ownvsls1	text	1
	ownmi	ownvsls2	text	1
	ownstate	ownvsls	text	2
	ownstate	ownvsls1	text	2
	ownstate	ownvsls2	text	2
	owntelno	ownvsls	text	12
	owntelno	ownvsls1	text	12
	owntelno	ownvsls2	text	12
	ownzip	ownvsls	text	9
	ownzip	ownvsls1	text	9
	ownzip	ownvsls2	text	9
	sailno	ownvsls2	text	8
	slcolor	ownvsls2	text	3
	trcolor	ownvsls2	text	3
	vhinown	ownvsls1	text	12

Used In	AttName	RelName	Type	Len
	vhinown	ownvsls2	text	12
	vslcall	ownvsls2	text	12
	vslconst	ownvsls2	text	1
	vsleng	ownvsls2	text	1
	vslfuel	ownvsls2	text	1
	vslhin	ownvsls	text	12
	vslhin	ownvsls2	text	12
	vslhp	ownvsls2	text	3
	vslhull	ownvsls2	text	1
	vsllenft	ownvsls2	integer	
	vsllenin	ownvsls2	integer	
	vslmake	ownvsls2	text	10
	vslmarks	ownvsls2	text	40
	vslmasts	ownvsls2	text	1
	vslmodel	ownvsls2	text	10
	vslname	ownvsls	text	17
	vslname	ownvsls2	text	17
	vslno	ownvsls2	text	8
	vsloffno	ownvsls2	text	10
	vslprop	ownvsls2	text	1
	vslsuper	ownvsls2	text	12
	vsltons	ownvsls2	integer	
	vsltype	ownvsls2	text	2
	vsluse	ownvsls2	text	2
	vslvalue	ownvsls2	dollar	
	vslyear	ownvsls2	text	4
perintel	intdob	persum	date	
	intfname	persum	text	15
	intlname	persum	text	20
	intmi	persum	text	1
	pintdate	persum	date	
	pintrem1	persum	text	70
	pintrem2	persum	text	70
	pintrem3	persum	text	70
quer_per	bodate	bohis	date	
	boname	bohis	text	25
	bono	bohis	text	7
	borate	bohis	text	5
	borem1	bohis	text	70
	borem2	bohis	text	70
	borem3	bohis	text	70
	botime	bohis	time	
	bovhin	bohis	text	12
	cbcolor	pvsljoin	text	3
	cbcolor	wvsljoin	text	3
	cmedecal	pvsljoin	text	1
	cmedecal	wvsljoin	text	1
	cmeyear	pvsljoin	text	4
	cmeyear	wvsljoin	text	4
	dkcolor	pvsljoin	text	3
	dkcolor	wvsljoin	text	3
	epiccode	vslint	text	1
	folderno	sarhis	text	7
	hlcolor	pvsljoin	text	3
	hlcolor	wvsljoin	text	3
	homeport	pvsljoin	text	25
	homeport	wvsljoin	text	25
	idoper	pvsljoin	integer	
	idoper	vslsopr	integer	
	idowner	vslsown	integer	
	idowner	wvsljoin	integer	
	immedsar	sarhis	text	1
	intdate	vslint	date	
	intdob	perint	date	
	intfname	perint	text	15

Used In	AttName	RelName	Type	Len
quer_per	intlname	perint	text	20
	intmi	perint	text	1
	intvname	vslint	text	17
	mucn	sarhis	text	8
	obslat	bohis	real	2.3
	obslong	bohis	real	3.3
	offshore	sarhis	integer	
	pintdate	perint	date	
	pintrem1	perint	text	70
	pintrem2	perint	text	70
	pintrem3	perint	text	70
	sailno	pvsljoin	text	8
	sailno	wvsljoin	text	8
	sardate	sarhis	date	
	sarlat	sarhis	real	2.3
	sarlong	sarhis	real	3.3
	sarnod	sarhis	text	25
	sarpob	sarhis	integer	
	sartime	sarhis	time	
	sarvhin	sarhis	text	12
	slcolor	pvsljoin	text	3
	slcolor	wvsljoin	text	3
	stolen	vslint	text	1
	tmp	temp	text	2
	trcolor	pvsljoin	text	3
	trcolor	wvsljoin	text	3
	ucn	sarhis	text	8
	uns69	bohis	text	1
	uns70	bohis	text	1
	uns71	bohis	text	1
	uns72	bohis	text	1
	uns73	bohis	text	1
	uns74	bohis	text	1
	uns75	bohis	text	1
	uns76	bohis	text	1
	uns77	bohis	text	1
	uns78	bohis	text	1
	uns79	bohis	text	1
	vhinoper	pvsljoin	text	12
	vhinoper	vslsopr	text	12
	vhinown	vslsown	text	12
	vhinown	wvsljoin	text	12
	viol54	bohis	text	1
	viol55	bohis	text	1
	viol56	bohis	text	1
	viol57	bohis	text	1
	viol58	bohis	text	1
	viol59	bohis	text	1
	viol60	bohis	text	1
	viol61	bohis	text	1
	viol62	bohis	text	1
	viol63	bohis	text	1
	viol64	bohis	text	1
	viol65	bohis	text	1
	viol66	bohis	text	1
	viol67	bohis	text	1
	viol68	bohis	text	1
	vremark1	vslint	text	70
	vremark2	vslint	text	70
	vremark3	vslint	text	70
	vslcall	pvsljoin	text	12
	vslcall	wvsljoin	text	12
	vslconst	pvsljoin	text	1
	vslconst	wvsljoin	text	1
	vsleng	pvsljoin	text	1

Used In	AttName	RelName	Type	Len
quer_per	vsleng	wvsljoin	text	1
	vslfuel	pvsljoin	text	1
	vslfuel	wvsljoin	text	1
	vslhin	pvsljoin	text	12
	vslhin	wvsljoin	text	12
	vslhp	pvsljoin	text	3
	vslhp	wvsljoin	text	3
	vslhull	pvsljoin	text	1
	vslhull	wvsljoin	text	1
	vsllenft	pvsljoin	integer	
	vsllenft	wvsljoin	integer	
	vsllenin	pvsljoin	integer	
	vsllenin	wvsljoin	integer	
	vslmake	pvsljoin	text	10
	vslmake	wvsljoin	text	10
	vslmarks	pvsljoin	text	40
	vslmarks	wvsljoin	text	40
	vslmasts	pvsljoin	text	1
	vslmasts	wvsljoin	text	1
	vslmodel	pvsljoin	text	10
	vslmodel	wvsljoin	text	10
	vslname	pvsljoin	text	17
	vslname	wvsljoin	text	17
	vslno	pvsljoin	text	8
	vslno	wvsljoin	text	8
	vsloffno	pvsljoin	text	10
	vsloffno	wvsljoin	text	10
	vslpob	bohis	integer	
	vslprop	pvsljoin	text	1
	vslprop	wvsljoin	text	1
	vslsuper	pvsljoin	text	12
	vslsuper	wvsljoin	text	12
	vsltons	pvsljoin	integer	
	vsltons	wvsljoin	integer	
	vsltype	pvsljoin	text	2
	vsltype	wvsljoin	text	2
	vsluse	pvsljoin	text	2
	vsluse	wvsljoin	text	2
	vslvalue	pvsljoin	dollar	
	vslvalue	wvsljoin	dollar	
	vslyear	pvsljoin	text	4
	vslyear	wvsljoin	text	4
	weapons	bohis	integer	
quer_vsl	bodate	bohis	date	
	boname	bohis	text	25
	bono	bohis	text	7
	borate	bohis	text	5
	borem1	bohis	text	70
	borem2	bohis	text	70
	borem3	bohis	text	70
	botime	bohis	time	
	bovhin	bohis	text	12
	epiccode	vslint	text	1
	folderno	sarhis	text	7
	immedsar	sarhis	text	1
	intdate	vslint	date	
	intdob	perint	date	
	intfname	perint	text	15
	intlname	perint	text	20
	intmi	perint	text	1
	intvname	vslint	text	17
	mucn	sarhis	text	8
	obslat	bohis	real	2.3
	obslong	bohis	real	3.3
	offshore	sarhis	integer	

Used In	AttName	RelName	Type	Len
quer_vsl	pintdate	perint	date	
	pintrem1	perint	text	70
	pintrem2	perint	text	70
	pintrem3	perint	text	70
	sardate	sarhis	date	
	sarlat	sarhis	real	2.3
	sarlong	sarhis	real	3.3
	sarnod	sarhis	text	25
	sarpob	sarhis	integer	
	sartime	sarhis	time	
	sarvhin	sarhis	text	12
	stolen	vslint	text	1
	tmp	temp	text	2
	ucn	sarhis	text	8
	uns69	bohis	text	1
	uns70	bohis	text	1
	uns71	bohis	text	1
	uns72	bohis	text	1
	uns73	bohis	text	1
	uns74	bohis	text	1
	uns75	bohis	text	1
	uns76	bohis	text	1
	uns77	bohis	text	1
	uns78	bohis	text	1
	uns79	bohis	text	1
	viol54	bohis	text	1
	viol55	bohis	text	1
	viol56	bohis	text	1
	viol57	bohis	text	1
	viol58	bohis	text	1
	viol59	bohis	text	1
	viol60	bohis	text	1
	viol61	bohis	text	1
	viol62	bohis	text	1
	viol63	bohis	text	1
	viol64	bohis	text	1
	viol65	bohis	text	1
	viol66	bohis	text	1
	viol67	bohis	text	1
	viol68	bohis	text	1
	vremark1	vslint	text	70
	vremark2	vslint	text	70
	vremark3	vslint	text	70
	vslpob	bohis	integer	
	weapons	bohis	integer	
sardump	ceiling	sarsum	integer	
	clouds	sarsum	integer	
	folderno	sarsum	text	7
	immedsar	sarsum	text	1
	mucn	sarsum	text	8
	offshore	sarsum	integer	
	sardate	sarsum	date	
	sarday	sarsum	text	3
	sarlat	sarsum	real	2.3
	sarlong	sarsum	real	3.3
	sarnod	sarsum	text	25
	sarpob	sarsum	integer	
	sartime	sarsum	time	
	sarvhin	sarsum	text	12
	seas	sarsum	integer	
	ssunrise	sarsum	time	
	ssunset	sarsum	time	
	swells	sarsum	integer	
	ucn	sarsum	text	8
	visdist	sarsum	integer	

Used In	AttName	RelName	Type	Len
sardump	winddir	sarsum	integer	
	windspd	sarsum	integer	
sarday	folderno	sarsum	text	7
	folderno	sarsum1	text	7
	pctincr	sarstat	real	
	sardate	sarsum	date	
	sardate	sarsum1	date	
	sarday	sarsum	text	3
	sarday	sarsum1	text	3
	sarnum	sarstat	integer	
	saryear	sarstat	text	4
	tmp	temp	text	2
sarmajor	folderno	sarsum	text	7
	freqcnt	positfrq	integer	
	gridpct	positfrq	real	
	majeast	positfrq	real	3.3
	majgrid	positfrq	text	2
	majnorth	positfrq	real	2.3
	majsouth	positfrq	real	2.3
	majwest	positfrq	real	3.3
	sarlat	sarsum	real	2.3
	sarlong	sarsum	real	3.3
sarminor	folderno	sarsum	text	7
	freqcnt	positfrq	integer	
	gridpct	positfrq	real	
	mineast	positfrq	real	3.3
	mingrid	positfrq	text	2
	minnorth	positfrq	real	2.3
	minwest	positfrq	real	3.3
	sarlat	sarsum	real	2.3
	sarlong	sarsum	real	3.3
sarsmbt	folderno	sarsum	text	7
	sarlat	sarsum	real	2.3
	sarlong	sarsum	real	3.3
	tmp	temp	text	2
sartimes	alngdate	sartime3	date	
	alngdate	sartime4	date	
	alngtime	sartime3	time	
	alngtime	sartime4	time	
	dvmdate	sartime5	date	
	dvmoored	sartime5	time	
	foldno	sartime1	text	7
	foldno	sartime2	text	7
	foldno	sartime3	text	7
	foldno	sartime4	text	7
	foldno	sartime5	text	7
	osdate	sartime1	date	
	osdate	sartime4	date	
	ostime	sartime1	time	
	ostime	sartime4	time	
	sutdate	sartime2	date	
	sutmoor	sartime2	time	
	tmp	temp	text	2
	towdate	sartime5	date	
	towtime	sartime5	time	
	uwdate	sartime1	date	
	uwdate	sartime2	date	
	uwdate	sartime3	date	
	uwtime	sartime1	time	
	uwtime	sartime2	time	
	uwtime	sartime3	time	
smboatrp	bono	bosum	text	7
	obslat	bosum	real	2.3
	obslong	bosum	real	3.3
	tmp	temp	text	2

Used In	AttName	RelName	Type	Len
vessel	cbcolor	vslsum	text	3
	cbcolor	vslsum1	text	3
	cbcolor	vslsum2	text	3
	cmedecal	vslsum	text	1
	cmedecal	vslsum1	text	1
	cmedecal	vslsum2	text	1
	cmeyear	vslsum	text	4
	cmeyear	vslsum1	text	4
	cmeyear	vslsum2	text	4
	dkcolor	vslsum	text	3
	dkcolor	vslsum1	text	3
	dkcolor	vslsum2	text	3
	hlcolor	vslsum	text	3
	hlcolor	vslsum1	text	3
	hlcolor	vslsum2	text	3
	homeport	vslsum	text	25
	homeport	vslsum1	text	25
	homeport	vslsum2	text	25
	sailno	vslsum	text	8
	sailno	vslsum1	text	8
	sailno	vslsum2	text	8
	slcolor	vslsum	text	3
	slcolor	vslsum1	text	3
	slcolor	vslsum2	text	3
	trcolor	vslsum	text	3
	trcolor	vslsum1	text	3
	trcolor	vslsum2	text	3
	vslcall	vslsum	text	12
	vslcall	vslsum1	text	12
	vslcall	vslsum2	text	12
	vslconst	vslsum	text	1
	vslconst	vslsum1	text	1
	vslconst	vslsum2	text	1
	vsleng	vslsum	text	1
	vsleng	vslsum1	text	1
	vsleng	vslsum2	text	1
	vslfuel	vslsum	text	1
	vslfuel	vslsum1	text	1
	vslfuel	vslsum2	text	1
	vslhin	vslsum	text	12
	vslhin	vslsum1	text	12
	vslhin	vslsum2	text	12
	vslhp	vslsum	text	3
	vslhp	vslsum1	text	3
	vslhp	vslsum2	text	3
	vslhull	vslsum	text	1
	vslhull	vslsum1	text	1
	vslhull	vslsum2	text	1
	vsllenft	vslsum	integer	
	vsllenft	vslsum1	integer	
	vsllenft	vslsum2	integer	
	vsllenin	vslsum	integer	
	vsllenin	vslsum1	integer	
	vsllenin	vslsum2	integer	
	vslmake	vslsum	text	10
	vslmake	vslsum1	text	10
	vslmake	vslsum2	text	10
	vslmarks	vslsum	text	40
	vslmarks	vslsum1	text	40
	vslmarks	vslsum2	text	40
	vslmasts	vslsum	text	1
	vslmasts	vslsum1	text	1
	vslmasts	vslsum2	text	1
	vslmodel	vslsum	text	10
	vslmodel	vslsum1	text	10

Used In	AttName	RelName	Type	Len
vessel	vslmodel	vslsum2	text	10
	vslname	vslsum	text	17
	vslname	vslsum1	text	17
	vslname	vslsum2	text	17
	vslno	vslsum	text	8
	vslno	vslsum1	text	8
	vslno	vslsum2	text	8
	vsloffno	vslsum	text	10
	vsloffno	vslsum1	text	10
	vsloffno	vslsum2	text	10
	vslprop	vslsum	text	1
	vslprop	vslsum1	text	1
	vslprop	vslsum2	text	1
	vslsuper	vslsum	text	12
	vslsuper	vslsum1	text	12
	vslsuper	vslsum2	text	12
	vsltons	vslsum	integer	
	vsltons	vslsum1	integer	
	vsltons	vslsum2	integer	
	vsltype	vslsum	text	2
	vsltype	vslsum1	text	2
	vsltype	vslsum2	text	2
	vsluse	vslsum	text	2
	vsluse	vslsum1	text	2
	vsluse	vslsum2	text	2
	vslvalue	vslsum	dollar	
	vslvalue	vslsum1	dollar	
	vslvalue	vslsum2	dollar	
	vslyear	vslsum	text	4
	vslyear	vslsum1	text	4
	vslyear	vslsum2	text	4
violfreq	bono	frqchart	text	7
	tmp	temp	text	2
	uns69	frqchart	text	1
	uns70	frqchart	text	1
	uns71	frqchart	text	1
	uns72	frqchart	text	1
	uns73	frqchart	text	1
	uns74	frqchart	text	1
	uns75	frqchart	text	1
	uns76	frqchart	text	1
	uns77	frqchart	text	1
	uns78	frqchart	text	1
	uns79	frqchart	text	1
	viol54	frqchart	text	1
	viol55	frqchart	text	1
	viol56	frqchart	text	1
	viol57	frqchart	text	1
	viol58	frqchart	text	1
	viol59	frqchart	text	1
	viol60	frqchart	text	1
	viol61	frqchart	text	1
	viol62	frqchart	text	1
	viol63	frqchart	text	1
	viol64	frqchart	text	1
	viol65	frqchart	text	1
	viol66	frqchart	text	1
	viol67	frqchart	text	1
	viol68	frqchart	text	1
vsloper	cbcolor	opvsls2	text	3
	cmedecal	opvsls2	text	1
	cmeyear	opvsls2	text	4
	dkcolor	opvsls2	text	3
	hlcolor	opvsls2	text	3
	homeport	opvsls2	text	25

Used In	AttName	RelName	Type	Len
vsloper	idoper	opvsls1	integer	
	idoper	opvsls2	integer	
	opaddr	opervsls	text	25
	opaddr	opvsls1	text	25
	opaddr	opvsls2	text	25
	opcity	opervsls	text	18
	opcity	opvsls1	text	18
	opcity	opvsls2	text	18
	opcourse	opvsls1	text	1
	opcourse	opvsls2	text	1
	opdob	opervsls	date	
	opdob	opvsls1	date	
	opdob	opvsls2	date	
	opfname	opervsls	text	15
	opfname	opvsls1	text	15
	opfname	opvsls2	text	15
	opid	opvsls1	integer	
	opid	opvsls2	integer	
	oplname	opervsls	text	20
	oplname	opvsls1	text	20
	oplname	opvsls2	text	20
	opmi	opervsls	text	1
	opmi	opvsls1	text	1
	opmi	opvsls2	text	1
	opstate	opervsls	text	2
	opstate	opvsls1	text	2
	opstate	opvsls2	text	2
	optelno	opervsls	text	12
	optelno	opvsls1	text	12
	optelno	opvsls2	text	12
	opzip	opervsls	text	9
	opzip	opvsls1	text	9
	opzip	opvsls2	text	9
	sailno	opvsls2	text	8
	slcolor	opvsls2	text	3
	trcolor	opvsls2	text	3
	vhinoper	opvsls1	text	12
	vhinoper	opvsls2	text	12
	vslcall	opvsls2	text	12
	vslconst	opvsls2	text	1
	vsleng	opvsls2	text	1
	vslfuel	opvsls2	text	1
	vslhin	opervsls	text	12
	vslhin	opvsls2	text	12
	vslhp	opvsls2	text	3
	vslhull	opvsls2	text	1
	vsllenft	opvsls2	integer	
	vsllenin	opvsls2	integer	
	vslmake	opvsls2	text	10
	vslmarks	opvsls2	text	40
	vslmasts	opvsls2	text	1
	vslmodel	opvsls2	text	10
	vslname	opervsls	text	17
	vslname	opvsls2	text	17
	vslno	opvsls2	text	8
	vsloffno	opvsls2	text	10
	vslprop	opvsls2	text	1
	vslsuper	opvsls2	text	12
	vsltons	opvsls2	integer	
	vsltype	opvsls2	text	2
	vsluse	opvsls2	text	2
	vslvalue	opvsls2	dollar	
	vslyear	opvsls2	text	4

Used In	AttName	RelName	Type	Len
vslowner	cbcolor	ownvsls2	text	3
	cmedecal	ownvsls2	text	1
	cmeyear	ownvsls2	text	4
	dkcolor	ownvsls2	text	3
	hlcolor	ownvsls2	text	3
	homeport	ownvsls2	text	25
	idowner	ownvsls1	integer	
	idowner	ownvsls2	integer	
	ownaddr	ownvsls	text	25
	ownaddr	ownvsls1	text	25
	ownaddr	ownvsls2	text	25
	owncity	ownvsls	text	18
	owncity	ownvsls1	text	18
	owncity	ownvsls2	text	18
	owndob	ownvsls	date	
	owndob	ownvsls1	date	
	owndob	ownvsls2	date	
	ownfname	ownvsls	text	15
	ownfname	ownvsls1	text	15
	ownfname	ownvsls2	text	15
	ownid	ownvsls1	integer	
	ownid	ownvsls2	integer	
	ownlname	ownvsls	text	20
	ownlname	ownvsls1	text	20
	ownlname	ownvsls2	text	20
	ownmi	ownvsls	text	1
	ownmi	ownvsls1	text	1
	ownmi	ownvsls2	text	1
	ownstate	ownvsls	text	2
	ownstate	ownvsls1	text	2
	ownstate	ownvsls2	text	2
	owntelno	ownvsls	text	12
	owntelno	ownvsls1	text	12
	owntelno	ownvsls2	text	12
	ownzip	ownvsls	text	9
	ownzip	ownvsls1	text	9
	ownzip	ownvsls2	text	9
	sailno	ownvsls2	text	8
	slcolor	ownvsls2	text	3
	trcolor	ownvsls2	text	3
	vhinown	ownvsls1	text	12
	vhinown	ownvsls2	text	12
	vslcall	ownvsls2	text	12
	vslconst	ownvsls2	text	1
	vsleng	ownvsls2	text	1
	vslfuel	ownvsls2	text	1
	vslhin	ownvsls	text	12
	vslhin	ownvsls2	text	12
	vslhp	ownvsls2	text	3
	vslhull	ownvsls2	text	1
	vsllenft	ownvsls2	integer	
	vsllenin	ownvsls2	integer	
	vslmake	ownvsls2	text	10
	vslmarks	ownvsls2	text	40
	vslmasts	ownvsls2	text	1
	vslmodel	ownvsls2	text	10
	vslname	ownvsls	text	17
	vslname	ownvsls2	text	17
	vslno	ownvsls2	text	8
	vsloffno	ownvsls2	text	10
	vslprop	ownvsls2	text	1
	vslsuper	ownvsls2	text	12
	vsltons	ownvsls2	integer	
	vsltype	ownvsls2	text	2
	vsluse	ownvsls2	text	2
	vslvalue	ownvsls2	dollar	
	vslyear	ownvsls2	text	4

Program Variables Sorted by Variable Name

Name	Type	Used In	Description
addl_opr	text	add_sbd.prg	user input
addl_own	text	add_sbd.prg	user input
addlownr	text	add_sar.prg add_sbd.prg	user input
after90	integer	cmerpt.prg	counting variable
algavg	real	sartimes.prg	average alongside time
algtimes	integer	sartimes.prg	counting variable
allrecs	text	opervsl.prg opervsls.prg ownvsl.prg ownvsl.prg vsloper.prg vsloper.prg vslowner.prg vslowner.prg	flag - all records?
anytype	text	vessel.prg	flag variable
aprdate1	date	sarday.prg	starting date
aprdate2	date	sarday.prg	ending date
aprsar	integer	sarday.prg	summation variable
augdate1	date	sarday.prg	starting date
augdate2	date	sarday.prg	ending date
augsar	integer	sarday.prg	summation variable
b1	date	add_brd.prg add_sbd.prg mod_brd.prg	bodate in boarding
b10	text	mod_brd.prg	obscity
b11	text	mod_brd.prg	obsstate in boarding
b12	real	mod_brd.prg	obslat in boarding
b13	real	mod_brd.prg	obslong in boarding
b14	text	mod_brd.prg	viol54 in boarding
b15	text	mod_brd.prg	viol55 in boarding
b16	text	mod_brd.prg	viol56 in boarding
b17	text	mod_brd.prg	viol57 in boarding
b18	text	mod_brd.prg	viol58 in boarding
b19	text	mod_brd.prg	viol59 in boarding
b2	time	add_brd.prg add_sbd.prg mod_brd.prg	botime in boarding
b20	text	mod_brd.prg	viol61 in boarding
b21	text	mod_brd.prg	viol61 in boarding
b22	text	mod_brd.prg	viol62 in boarding
b23	text	mod_brd.prg	viol63 in boarding
b24	text	mod_brd.prg	viol64 in boarding
b25	text	mod_brd.prg	viol65 in boarding
b26	text	mod_brd.prg	viol66 in boarding
b27	text	mod_brd.prg	viol67 in boarding
b28	text	mod_brd.prg	viol68 in boarding
b29	text	mod_brd.prg	uns69 in boarding
b3	text	mod_brd.prg	bono in boarding
	real	add_brd.prg add_sbd.prg	obslat in boarding
b30	text	mod_brd.prg	uns70 in boarding
b31	text	mod_brd.prg	uns71 in boarding
b32	text	mod_brd.prg	uns72 in boarding
b33	text	mod_brd.prg	uns73 in boarding
b34	text	mod_brd.prg	uns74 in boarding
b35	text	mod_brd.prg	uns75 in boarding
b36	text	mod_brd.prg	uns76 in boarding
b37	text	mod_brd.prg	uns77 in boarding
b38	text	mod_brd.prg	uns78 in boarding
b39	text	mod_brd.prg	uns79 in boarding

Name	Type	Used In	Description
b4	integer	mod_brd.prg	vslpob in boarding
	real	add_brd.prg add_sbd.prg	obslong in boarding
b40	text	mod_brd.prg	borem1 in boarding
b41	text	mod_brd.prg	borem2 in boarding
b42	text	mod_brd.prg	borem3 in boarding
b43	text	mod_brd.prg	boname in boarding
b44	text	mod_brd.prg	borate in boarding
b45	text	mod_brd.prg	bounit in boarding
b46	text	mod_brd.prg	bodist in boarding
b47	text	mod_brd.prg	boopfac in boarding
b48	text	mod_brd.prg	boday in boarding
b49	time	mod_brd.prg	bsunrise in boarding
b5	integer	mod_brd.prg	adultpfd in boarding
b50	time	mod_brd.prg	bsunset in boarding
b51	text	mod_brd.prg	bovhin in boarding
b52	integer	mod_brd.prg	weapons in boarding
b6	integer	mod_brd.prg	childpfd in boarding
b7	text	mod_brd.prg	owstatus in boarding
b8	text	mod_brd.prg	obsbody in boarding
b9	text	mod_brd.prg	obscount in boarding
backup	text	add_brd.prg add_sar.prg add_sbd.prg	flag variable
before70	integer	cmerpt.prg	counting variable
bohisprn	text	quer_per.prg quer_vsl.prg	user input
both	text	mod_sut.prg	flag variable
chances	integer	startup.prg	chances to logon
change	text	mod_sar.prg mod_sut.prg sarday.prg	flag variable
chk_hin	text	mod_brd.prg mod_vhin.prg	flag variable
chk_name	text	mod_brd.prg mod_vhin.prg	flag variable
chk_ownr	text	mod_owop.prg	flag variable
chk_hin	text	quer_vsl.prg	flag variable
chk_intl	text	quer_vsl.prg	flag variable
chk_name	text	quer_vsl.prg	flag variable
chk_no	text	quer_vsl.prg	flag variable
chk_op	text	quer_per.prg	flag variable
chkoffno	text	quer_vsl.prg	flag variable
choice	text	add_brd.prg add_sar.prg add_sbd.prg am_pint.prg am_vint.prg mod_owop.prg mod_owop.prg packdbms.prg quer_vsl.prg vessel.prg	user input variable
cmecount	integer	cmerpt.prg	counting variable
code	integer	intelrpt.prg	input variable
constrds	text	intelrpt.prg	= consdesc in CONSTR
consttst	text	intelrpt.prg	= constype in CONSTR
cont	text	add_brd.prg add_sar.prg add_sbd.prg	user input
correct	text	am_pint.prg am_vint.prg mod_brd.prg mod_sar.prg mod_sut.prg	user input

Name	Type	Used In	Description
correct	text	mod_vhin.prg mod_vhin.prg	user input
count	integer	quer_per.prg quer_vsl.prg sarday.prg startup.prg	counting variable
datadiff	integer	bominor.prg	date difference
datdiff	integer	bomajor.prg	date difference
datediff	integer	boarding.prg bominor.prg cmerpt.prg daynight.prg epicdump.prg immedsar.prg intelrpt.prg operator.prg owners.prg perintel.prg sardump.prg sarmajor.prg sarminor.prg sarsmbt.prg sartimes.prg smboatrp.prg violfreq.prg	date difference
datedour	real	sartimes.prg	time computation variable
daybrd	integer	daynight.prg	counting variable
daysar	integer	daynight.prg	counting variable
decdate1	date	sarday.prg	starting date
decdate2	date	sarday.prg	ending date
decsar	integer	sarday.prg	summation variable
difbo	text	add_sbd.prg add_brd.prg	flag variable
difsar	text	add_sar.prg add_sbd.prg	flag variable
done	text	add_ownr.prg add_sut.prg mod_sut.prg quer_per.prg	flag variable
easttst	real	bomajor.prg bominor.prg sarmajor.prg sarminor.prg	nautical position coord.
edsc1	text	epicdump.prg	= epicdsc1 in VSLINTEL
edsc2	text	epicdump.prg	= epicdsc2 in VSLINTEL
edsc3	text	epicdump.prg	= epicdsc3 in VSLINTEL
edsc4	text	epicdump.prg	= epicdsc4 in VSLINTEL
enddate	date	boarding.prg bomajor.prg bominor.prg cmerpt.prg daynight.prg epicdump.prg immedsar.prg	ending date

Name	Type	Used In	Description
enddate	date	intelsum.prg operator.prg owners.prg perintel.prg sarday.prg sardump.prg sarmajor.prg sarminor.prg sarsmbt.prg sartimes.prg smboatrp.prg violfreq.prg	ending date
endltr	text	operator.prg opervsl.prg opervsls.prg owners.prg ownvsl.prg perintel.prg vessel.prg vsloper.prg vslowner.prg	ending letter
engdescr	text	intelrpt.prg	= ecomptyp in ENGCOMP
engtst	text	intelrpt.prg	= ecomptype in ENGCOMP
epicnum	integer	epicdump.prg	counting variable
epictest	text	epicdump.prg	= epiccode in VSLINTEL
er	text	mod_brd.prg mod_brd.prg mod_owop.prg mod_sar.prg mod_vhin.prg	user input
error	text	mod_brd.prg mod_owop.prg mod_owop.prg mod_vhin.prg	flag variable
fdesc	text	intelrpt.prg	= fcompdes in FUELCOMP
febdate1	date	sarday.prg	starting date
febdate2	date	sarday.prg	ending date
febsar	integer	sarday.prg	summation variable
fixopr	text	add_brd.prg add_sar.prg add_sbd.prg	flag variable
fixown	text	add_brd.prg add_sar.prg add_sbd.prg	flag variable
fixvsl	text	am_vint.prg	user input
foldok	text	mod_sut.prg	flag variable
found	text	sarday.prg	flag variable
foundv	text	mod_vhin.prg	user input
friday	integer	sarday.prg	counting variable
fueltst	text	intelrpt.prg	= fcomptyp in FUELCOMP
go	text	mod_owop.prg	user input
gridcnt	integer	sarmajor.prg sarminor.prg	counting variable
hdesc	text	intelrpt.prg	= hmatdesc in HULLMAT
hulltst	text	intelrpt.prg	= hmattype in HULLMAT
inarea	integer	bomajor.prg bominor.prg sarmajor.prg sarminor.prg	summation variable
initdate	date	sarday.prg	starting date
inpct	real	bomajor.prg bominor.prg sarmajor.prg sarminor.prg sarsmbt.prg	percentage

Name	Type	Used In	Description
inpcttmp	real	sarsmbt.prg	percentage
inside	integer	sarsmbt.prg smboatrp.prg	summation variable
jandate1	date	sarday.prg	starting date
jandate2	date	sarday.prg	ending date
jansar	integer	sarday.prg	summation variable
juldate1	date	sarday.prg	starting date
juldate2	date	sarday.prg	ending date
julsar	integer	sarday.prg	summation variable
jundate1	date	sardates.prg sarday.prg	time computation variable
jundate2	date	sarday.prg	ending date
junsar	integer	sarday.prg	summation variable
legalch	text	vessel.prg	flag variable
legaltyp	text	vessel.prg	flag variable
legaluse	text	vessel.prg	flag variable
len2030	integer	intelrpt.prg	counting variable
len3040	integer	intelrpt.prg	counting variable
len4050	integer	intelrpt.prg	counting variable
len5060	integer	intelrpt.prg	counting variable
len6070	integer	intelrpt.prg	counting variable
len7080	integer	intelrpt.prg	counting variable
len8090	integer	intelrpt.prg	counting variable
len90100	integer	intelrpt.prg	counting variable
len<20	integer	intelrpt.prg	counting variable
len>100	integer	intelrpt.prg	counting variable
loadmenu	text	add_brd.prg add_sar.prg add_sbd.prg	flag variable
loadok	text	add_ownr.prg	flag variable
ltrdiff	integer	operator.prg opervsl.prg opervsls.prg owners.prg ownvsl.prg perintel.prg vessel.prg vsloper.prg vslowner.prg	letter difference
mardate1	date	sarday.prg	starting date
mardate2	date	sarday.prg	ending date
marsar	integer	sarday.prg	summation variable
maxaltim	real	sartimes.prg	maximum alongside time
maxostim	real	sartimes.prg	maximum on scene time
maxshtim	real	sartimes.prg	maximum search time
maxtttim	real	sartimes.prg	maximum total case time
maxtwtim	real	sartimes.prg	maximum towing time
maydate1	date	sarday.prg	starting date
maydate2	date	sarday.prg	ending date
maysar	integer	sarday.prg	summation variable
menu	text	quer_vsl.prg	flag variable
minaltim	real	sartimes.prg	minimum alongside time
minostim	real	sartimes.prg	minimum on scene time
minshtim	real	sartimes.prg	minimum search time
mintttim	real	sartimes.prg	minimum total case time
mintwtim	real	sartimes.prg	minimum towing time
monday	integer	sarday.prg	counting variable
more	text	add_sbd.prg add_brd.prg add_ownr.prg add_sar.prg am_pint.prg am_vint.prg mod_brd.prg	flag variable

Name	Type	Used In	Description
more	text	mod_sar.prg mod_sut.prg mod_vhin.prg quer_per.prg quer_vsl.prg	flag variable
mrintel	text	am_pint.prg am_vint.prg	user input
mrname	text	quer_per.prg	user input
mropers	text	add_brd.prg add_sbd.prg	user input
mrownrs	text	add_brd.prg add_ownr.prg add_sar.prg add_sbd.prg	user input
mrsar	text	mod_sar.prg	user input
mrsut	text	mod_sut.prg	user input
mrunits	text	add_sar.prg add_sbd.prg add_sut.prg	user input
mrvsls	text	mod_brd.prg mod_vhin.prg	user input
name	text	mod_sut.prg	flag variable
nameok	text	quer_vsl.prg	flag variable
new	integer	sarday.prg	this year's sar cases
nitebrd	integer	daynight.prg	counting variable
nitesar	integer	daynight.prg	counting variable
nmchange	text	mod_owop.prg	flag variable
northtst	real	bomajor.prg bominor.prg sarmajor.prg sarminor.prg	nautical position coord.
notimmed	integer	immedsar.prg	non-immediate SAR Cases
novdate1	date	sarday.prg	starting date
novdate2	date	sarday.prg	ending date
novsar	integer	sarday.prg	summation variable
numrows	integer	boarding.prg immedsar.prg intelrpt.prg operator.prg opervsls.prg owners.prg ownvsl.prg perintel.prg sarday.prg sardump.prg sarmajor.prg sarminor.prg sarsmbt.prg vessel.prg vsloper.prg vslowner.prg	counting variable
numyears	integer	sarday.prg	counting variable
obslat	real	smboatrp.prg	nautical position coord.
obslong	real	smboatrp.prg	nautical position coord.
octdate1	date	sarday.prg	starting date
octdate2	date	sarday.prg	ending date
octsar	integer	sarday.prg	summation variable
old	integer	sarday.prg	previous year's sar cases
oldfold	text	mod_sar.prg	folder number
op1	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	op1name in operator

Name	Type	Used In	Description
op10	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	opcourse in operator
op11	integer	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	opid in operator
op2	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	opfname in operator
op3	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	opmi in operator
op4	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	opaddr in operator
op5	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	opcity in operator
op6	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	opstate in operator
op7	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	opzip in operator
op8	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	optelno in operator
op9	date	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	opdob in operator
opchange	text	mod_owop.prg	flag variable
opfound	text	mod_owop.prg	flag variable
opid	integer	mod_owop.prg quer_per.prg quer_vsl.prg	operator id (system)
oprfound	text	quer_per.prg	flag variable
oscavg	real	sartimes.prg	average on scene time
osdate	date	sartimes.prg	on scene date
ossum	real	sartimes.prg	summation variable
ostime	time	sartimes.prg	on scene time
ostimes	integer	sartimes.prg	counting variable
out	text	add_ownr.prg	flag variable
outarea	integer	bomajor.prg bominor.prg sarmajor.prg sarminor.prg	summation variable
outpct	real	bomajor.prg bominor.prg sarmajor.prg sarminor.prg sarsmbt.prg	percentage
outpcttm	real	sarsmbt.prg	percentage
outside	integer	sarsmbt.prg smboatrp.prg	summation variable

Name	Type	Used In	Description
ow1	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	ownlname in owners
ow10	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	ownid in owners
ow2	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	ownfname in owners
ow3	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	ownmi in owners
ow4	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	ownaddr in owners
ow5	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	owncity in owners
ow6	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	ownstate in owners
ow7	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	ownzip in owners
ow8	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	owntelno in owners
ow9	date	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	owndob in owners
owchange	text	mod_owop.prg	flag variable
owfound	text	mod_owop.prg	flag variable
owhisprn	text	quer_vsl.prg	user input
owid	integer	quer_vsl.prg	owner id (system)
owndone	text	quer_per.prg	flag variable
ownfound	text	quer_per.prg	flag variable
owoptest	text	quer_per.prg	flag variable
pcik6	integer	brd_main.prg	counting variable
pct2030	real	intelrpt.prg	percentage
pct3040	real	intelrpt.prg	percentage
pct4050	real	intelrpt.prg	percentage
pct5060	real	intelrpt.prg	percentage
pct54	real	violfreq.prg	percentage
pct54tmp	real	violfreq.prg	percentage
pct55	real	violfreq.prg	percentage
pct55tmp	real	violfreq.prg	percentage
pct56	real	violfreq.prg	percentage
pct56tmp	real	violfreq.prg	percentage
pct57	real	violfreq.prg	percentage
pct57tmp	real	violfreq.prg	percentage
pct58	real	violfreq.prg	percentage
pct58tmp	real	violfreq.prg	percentage
pct59	real	violfreq.prg	percentage
pct59tmp	real	violfreq.prg	percentage
pct60	real	violfreq.prg	percentage

Name	Type	Used In	Description
pct6070	real	intelrpt.prg	percentage
pct60tmp	real	violfreq.prg	percentage
pct61	real	violfreq.prg	percentage
pct61tmp	real	violfreq.prg	percentage
pct62	real	violfreq.prg	percentage
pct62tmp	real	violfreq.prg	percentage
pct63	real	violfreq.prg	percentage
pct63tmp	real	violfreq.prg	percentage
pct64	real	violfreq.prg	percentage
pct64tmp	real	violfreq.prg	percentage
pct65	real	violfreq.prg	percentage
pct65tmp	real	violfreq.prg	percentage
pct66	real	violfreq.prg	percentage
pct66tmp	real	violfreq.prg	percentage
pct67	real	violfreq.prg	percentage
pct67tmp	real	violfreq.prg	percentage
pct68	integer	violfreq.prg	percentage
pct68tmp	real	violfreq.prg	percentage
pct69	real	violfreq.prg	percentage
pct69tmp	real	violfreq.prg	percentage
pct70	real	violfreq.prg	percentage
pct7080	real	intelrpt.prg	percentage
pct70tmp	real	violfreq.prg	percentage
pct71	real	violfreq.prg	percentage
pct7175	real	cmerpt.prg	percentage
pct71tmp	real	violfreq.prg	percentage
pct72	real	violfreq.prg	percentage
pct72tmp	real	violfreq.prg	percentage
pct73	real	violfreq.prg	percentage
pct73tmp	real	violfreq.prg	percentage
pct74	real	violfreq.prg	percentage
pct74tmp	real	violfreq.prg	percentage
pct75	real	violfreq.prg	percentage
pct75tmp	real	violfreq.prg	percentage
pct76	real	violfreq.prg	percentage
pct7679	real	cmerpt.prg	percentage
pct76tmp	real	violfreq.prg	percentage
pct77	real	violfreq.prg	percentage
pct77tmp	real	violfreq.prg	percentage
pct78	real	violfreq.prg	percentage
pct78tmp	real	violfreq.prg	percentage
pct79	real	violfreq.prg	percentage
pct79tmp	real	violfreq.prg	percentage
pct8081	real	cmerpt.prg	percentage
pct8090	real	intelrpt.prg	percentage
pct8283	real	cmerpt.prg	percentage
pct8485	real	cmerpt.prg	percentage
pct8687	real	cmerpt.prg	percentage
pct8889	real	cmerpt.prg	percentage
pct9010	real	intelrpt.prg	percentage
pct90100	real	intelrpt.prg	percentage
pct<00	real	intelrpt.prg	percentage
pct<20	real	intelrpt.prg	percentage
pct<60	real	intelrpt.prg	percentage
pct<65	real	intelrpt.prg	percentage
pct<70	real	intelrpt.prg	percentage
pct<75	real	intelrpt.prg	percentage
pct<80	real	intelrpt.prg	percentage
pct<85	real	intelrpt.prg	percentage
pct<90	real	intelrpt.prg	percentage
pct<95	real	intelrpt.prg	percentage
pct>00	real	intelrpt.prg	percentage
pct>100	real	intelrpt.prg	percentage
pctapr	real	sarday.prg	percentage
pctaug	real	sarday.prg	percentage

Name	Type	Used In	Description
pctchang	real	sarday.prg	percentage
pctcme	real	cmerpt.prg	percentage
pctdayb	real	daynight.prg	percentage
pctdays	real	daynight.prg	percentage
pctdec	real	sarday.prg	percentage
pctfeb	real	sarday.prg	percentage
pctfri	real	sarday.prg	percentage
pctgt90	real	cmerpt.prg	percentage
pctimmed	real	immedsar.prg	% immediate SAR cases
pctjan	real	sarday.prg	percentage
pctjul	real	sarday.prg	percentage
pctjun	real	sarday.prg	percentage
pctlt70	real	cmerpt.prg	percentage
pctmar	real	sarday.prg	percentage
pctmay	real	sarday.prg	percentage
pctmon	real	sarday.prg	percentage
pctniteb	real	daynight.prg	percentage
pctnites	real	daynight.prg	percentage
pctnotim	real	immedsar.prg	% non-immediate SAR Cases
pctnov	real	sarday.prg	percentage
pctoct	real	sarday.prg	percentage
pctsat	real	sarday.prg	percentage
pctsep	real	sarday.prg	percentage
pctsun	real	sarday.prg	percentage
pcttemp1	real	epicdump.prg	% epiccode / total vsls
pcttemp2	real	epicdump.prg	% epiccode / total vsls
pctthu	real	sarday.prg	percentage
pcttue	real	sardav.prg	percentage
pctwed	real	sarday.prg	percentage
perintpr	text	quer_per.prg	user input
persprn	text	quer_per.prg	user input
pg	integer	add_brd.prg add_sar.prg add_sbd.prg mod_brd.prg	counting variable
pgcount	integer	add_brd.prg add_sar.prg add_sbd.prg	counting variable
pick1	integer	sarsys.prg	menu variable
pick10	integer	qrymain.prg	menu variable
pick11	integer	utlmain.prg	menu variable
pick2	integer	inpmain.prg	menu variable
pick3	integer	modmain.prg	menu variable
pick4	integer	rptmain.prg	menu variable
pick5	integer	int_main.prg	menu variable
pick7	integer	sar_main.prg	menu variable
pick8	integer	vsl_main.prg	menu variable
pick9	integer	prs_main.prg	menu variable
prdesc	text	intelrpt.prg	= propdesc in BOATPROP
proptst	text	intelrpt.prg	= proptype in BOATPROP
reenter	text	add_ownr.prg am_pint.prg am_vint.prg mod_sar.prg mod_sut.prg	user input
review	text	add_brd.prg add_sar.prg add_sbd.prg mod_brd.prg	flag variable
sunits	text	add_sar.prg add_sbd.prg	user input

Name	Type	Used In	Description
s1	date	add_sar.prg add_sbd.prg	sardate in sar
	text	mod_sar.prg	folderno in sar
s10	integer	mod_sar.prg	sarpob in sar
s11	text	mod_sar.prg	sarvhin in sar
s12	real	mod_sar.prg	sarlat in sar
s13	real	mod_sar.prg	sarlong in sar
s14	integer	mod_sar.prg	distoff in sar
s15	text	mod_sar.prg	immedsar in sar
s16	text	mod_sar.prg	winddir in sar
s17	text	mod_sar.prg	windspd in sar
s18	text	mod_sar.prg	visdist in sar
s19	text	mod_sar.prg	clouds in sar
s2	time	add_sar.prg add_sbd.prg	sartime in sar
	text	mod_sar.prg	ucn in sar
s20	text	mod_sar.prg	ceiling in sar
s21	text	mod_sar.prg	seas in sar
s22	text	mod_sar.prg	swells in sar
s3	real	add_sar.prg add_sbd.prg	sarlat in sar
	text	mod_sar.prg	mucn in sar
s4	real	add_sar.prg add_sbd.prg	sarlong in sar
	date	mod_sar.prg	sardate in sar
s5	text	add_sar.prg add_sbd.prg	sarnod in sar
	text	mod_sar.prg	sarday in sar
s6	text	add_sar.prg add_sbd.prg	sarvhin in sar
	time	mod_sar.prg	sartime in sar
s7	time	mod_sar.prg	ssunrise in sar
s8	time	mod_sar.prg	ssunset
s9	text	mod_sar.prg	sarnod in sar
sar00	integer	sarday.prg	counting variable
sar01	integer	sarday.prg	counting variable
sar02	integer	sarday.prg	counting variable
sar03	integer	sarday.prg	counting variable
sar04	integer	sarday.prg	counting variable
sar05	integer	sarday.prg	counting variable
sar06	integer	sarday.prg	counting variable
sar07	integer	sarday.prg	counting variable
sar08	integer	sarday.prg	counting variable
sar09	integer	sarday.prg	counting variable
sar10	integer	sarday.prg	counting variable
sar11	integer	sarday.prg	counting variable
sar12	integer	sarday.prg	counting variable
sar81	integer	sarday.prg	counting variable
sar82	integer	sarday.prg	counting variable
sar83	integer	sarday.prg	counting variable
sar84	integer	sarday.prg	counting variable
sar85	integer	sarday.prg	counting variable
sar86	integer	sarday.prg	counting variable
sar87	integer	sarday.prg	counting variable
sar88	integer	sarday.prg	counting variable
sar89	integer	sarday.prg	counting variable
sar90	integer	sarday.prg	counting variable
sar91	integer	sarday.prg	counting variable
sar92	integer	sarday.prg	counting variable
sar93	integer	sarday.prg	counting variable
sar94	integer	sarday.prg	counting variable
sar95	integer	sarday.prg	counting variable
sar96	integer	sarday.prg	counting variable
sar97	integer	sarday.prg	counting variable
sar98	integer	sarday.prg	counting variable

Name	Type	Used In	Description
sar99	integer	sarday.prg	counting variable
sarhsprn	text	quer_per.prg quer_vsl.prg	user input
saturday	integer	sarday.prg	counting variable
schavg	real	sartimes.prg	average search time
schtimes	integer	sartimes.prg	counting variable
searchok	text	quer_per.prg	flag variable
sepdate1	date	sarday.prg	starting date
sepdate2	date	sarday.prg	ending date
sepsar	integer	sarday.prg	summation variable
southtst	real	bomajor.prg bominor.prg sarmajor.prg sarminor.prg	nautical position coord.
status	text	quer_per.prg	flag variable
strtdate	date	boarding.prg bomajor.prg bominor.prg cmerpt.prg daynight.prg epicdump.prg immedsar.prg intelrpt.prg operator.prg owners.prg perintel.prg sardump.prg sarmajor.prg sarminor.prg sarsmbt.prg sartimes.prg smboatrp.prg violfreq.prg	starting date
strtltr	text	operator.prg opervsl.prg opervsls.prg owners.prg ownvsl.prg perintel.prg vessel.prg vsloper.prg vslowner.prg	starting letter
su1	text	mod_sut.prg	foldno in sarunits
su10	date	mod_sut.prg	alngdate in sarunits
su11	time	mod_sut.prg	towtime in sarunits
su12	date	mod_sut.prg	towdate in sarunits
su13	time	mod_sut.prg	dvmoored in sarunits
su14	date	mod_sut.prg	dvmdate in sarunits
su15	time	mod_sut.prg	sutmoor in sarunits
su16	date	mod_sut.prg	sutdate in sarunits
su2	text	mod_sut.prg	unit in sarunits
su3	time	mod_sut.prg	uwtime in sarunits
su4	date	mod_sut.prg	uwdate
su5	time	mod_sut.prg	oseta in sarunits
su6	date	mod_sut.prg	etadate in sarunits
su7	time	mod_sut.prg	ostime in sarunits
su8	date	mod_sut.prg	osdate in sarunits
su9	text	mod_sut.prg	alngtime in sarunits
sum1	integer	sarday.prg	summation variable
sum10	integer	sarday.prg	summation variable
sum11	integer	sarday.prg	summation variable
sum12	integer	sarday.prg	summation variable
sum2	integer	sarday.prg	summation variable
sum20	integer	sarday.prg	summation variable
sum3	integer	sarday.prg	summation variable

Name	Type	Used In	Description
sum4	integer	sarday.prg	summation variable
sum5	integer	sarday.prg	summation variable
sum6	integer	sarday.prg	summation variable
sumconst	integer	intelrpt.prg	counting variable
sumeng	integer	intelrpt.prg	counting variable
sumfuel	integer	intelrpt.prg	counting variable
sumhull	integer	intelrpt.prg	counting variable
sumprop	integer	intelrpt.prg	counting variable
sumtype	integer	intelrpt.prg	counting variable
sumuse	integer	intelrpt.prg	counting variable
sunday	integer	sarday.prg	counting variable
sutdate	date	sartimes.prg	sar unit moored date
sutmoor	time	sartimes.prg	sar unit moored time
temppct	real	bomajor.prg bominor.prg sarmajor.prg sarminor.prg	percentage
test	real	sartimes.prg	comparison variable
thursday	integer	sarday.prg	counting variable
timeadd	real	sartimes.prg	time computation variable
timehour	real	sartimes.prg	time computation variable
totavg	real	sartimes.prg	average total case time
totbrds	integer	cmerpt.prg daynight.prg intelrpt.prg smboatrp.prg	counting variable
totepics	integer	epicdump.prg	counting variable
totimmed	integer	immedsar.prg	counting variable
totsars	integer	daynight.prg immedsar.prg sarday.prg sarmajor.prg sarsmbt.prg	total sars
totsum	real	sartimes.prg	summation variable
tottimes	integer	sartimes.prg	counting variable
totviols	integer	intelrpt.prg intelrpt.prg	summation variable
towavg	real	sartimes.prg	average towing time
towsum	real	sartimes.prg	summation variable
towtimes	integer	sartimes.prg	counting variable
tuesday	integer	sarday.prg	counting variable
tv1	integer	intelrpt.prg	summation variable
tv10	integer	intelrpt.prg	summation variable
tv2	integer	intelrpt.prg	summation variable
tv20	integer	intelrpt.prg	summation variable
tv3	integer	intelrpt.prg	summation variable
tv30	integer	intelrpt.prg	summation variable
tv4	integer	intelrpt.prg	summation variable
tv5	integer	intelrpt.prg	summation variable
typchoic	text	vessel.prg	user type choice
typdesc1	text	intelrpt.prg	= vtypedes in BOATTYPE
typetest	text	intelrpt.prg vessel.prg	= vtype in BOATTYPE
uns69s	integer	violfreq.prg	counting variable
uns70s	integer	violfreq.prg	counting variable
uns71s	integer	violfreq.prg	counting variable
uns72s	integer	violfreq.prg	counting variable
uns73s	integer	violfreq.prg	counting variable
uns74s	integer	violfreq.prg	counting variable
uns75s	integer	violfreq.prg	counting variable
uns76s	integer	violfreq.prg	counting variable
uns77s	integer	violfreq.prg	counting variable
uns78s	integer	violfreq.prg	counting variable
uns79s	integer	violfreq.prg	counting variable
unstotl	integer	violfreq.prg	summation variable

Name	Type	Used In	Description
unstot2	integer	violfreq.prg	summation variable
unstot3	integer	violfreq.prg	summation variable
unstot4	integer	violfreq.prg	summation variable
unstot5	integer	violfreq.prg	summation variable
unstot6	integer	violfreq.prg	summation variable
unstot7	integer	violfreq.prg	summation variable
unstotal	integer	violfreq.prg	total unsafe conditions
usechoic	text	vessel.prg	user vessel use choice
usedesc1	text	intelrpt.prg	= vusedesc in BOATUSE
username	text	startup.prg	user name input
userpass	text	startup.prg	user password
usetest	text	intelrpt.prg vessel.prg	= vuse in BOATUSE
uwdate	date	sartimes.prg	underway date
uwtime	time	sartimes.prg	underway time
v1	text	add_brd.prg add_sar.prg add_sbd.prg	vslno in vessel
v10	text	add_brd.prg add_sar.prg add_sbd.prg	vslhp in vessel
v11	text	add_brd.prg add_sar.prg add_sbd.prg	vsluse in vessel
v12	text	add_brd.prg add_sar.prg add_sbd.prg	vsltype in vessel
v13	text	add_brd.prg add_sar.prg add_sbd.prg	vslprop in vessel
v14	text	add_brd.prg add_sar.prg add_sbd.prg	vslhull in vessel
v15	text	add_brd.prg add_sar.prg add_sbd.prg	vsleng in vessel
v16	text	add_brd.prg add_sar.prg add_sbd.prg	vslfuel in vessel
v17	text	add_brd.prg add_sar.prg add_sbd.prg	vslconst in vessel
v18	text	add_brd.prg add_sar.prg add_sbd.prg	cmedecal in vessel
v19	text	add_brd.prg add_sar.prg add_sbd.prg	cmeyuear in vessel
v2	text	add_brd.prg add_sar.prg add_sbd.prg	vslname in vessel
v20	text	add_brd.prg add_sar.prg add_sbd.prg	vsloffno in vessel
v21	text	add_brd.prg add_sar.prg add_sbd.prg	homeport in vessel
v22	text	add_brd.prg add_sar.prg add_sbd.prg	vslcall in vessel
v23	text	add_brd.prg add_sar.prg add_sbd.prg	vslvalue in vessel

Name	Type	Used In	Description
v24	text	add_brd.prg add_sar.prg add_sbd.prg	vslsuper in vessel
v25	text	add_brd.prg add_sar.prg add_sbd.prg	vslmasts in vessel
v26	text	add_brd.prg add_sar.prg add_sbd.prg	sailno in vessel
v27	text	add_brd.prg add_sar.prg add_sbd.prg	hlcolor in vessel
v28	text	add_brd.prg add_sar.prg add_sbd.prg	trcolor in vessel
v29	text	add_brd.prg add_sar.prg add_sbd.prg	slcolor in vessel
v3	text	add_brd.prg add_sar.prg add_sbd.prg	vslhin in vessel
v30	text	add_brd.prg add_sar.prg add_sbd.prg	cbcolor in vessel
v31	text	add_brd.prg add_sar.prg add_sbd.prg	dkcolor in vessel
v32	text	add_brd.prg add_sar.prg add_sbd.prg	vslmarks in vessel
v4	text	add_brd.prg add_sar.prg add_sbd.prg	vslmake in vessel
v5	text	add_brd.prg add_sar.prg add_sbd.prg	vslmodel in vessel
v6	text	add_brd.prg add_sar.prg add_sbd.prg	vslyear in vessel
v7	integer	add_brd.prg add_sar.prg add_sbd.prg	vsltons in vessel
v8	integer	add_brd.prg add_sar.prg add_sbd.prg	vsllenft in vessel
v9	integer	add_brd.prg add_sar.prg add_sbd.prg	vsllenin in vessel
vadult	integer	add_brd.prg add_sbd.prg mod_brd.prg	adultpfd in boarding
valc	text	quer_per.prg quer_vsl.prg	operator alcohol conditio
valgdate	date	add_sar.prg add_sbd.prg add_sut.prg mod_sut.prg	date alongside
valgtime	time	add_sar.prg add_sbd.prg add_sut.prg mod_sut.prg	time alongside
validusr	text	startup.prg	flag variable
vbadhin	text	mod_vhin.prg	vslhin in vessel

Name	Type	Used In	Description
vbodate	date	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	boarding date
vboday	text	add_brd.prg add_sbd.prg mod_brd.prg	boday in boarding
vbodist	text	add_brd.prg add_sbd.prg mod_brd.prg	bodist in boarding
vbolat	real	quer_per.prg quer_vsl.prg	latitude position
vbolong	real	quer_per.prg quer_vsl.prg	longitude position
vboname	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	boname in boarding
vbono	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	boarding report number
vboopfac	text	add_brd.prg add_sbd.prg mod_brd.prg	boopfac in boarding
vborate	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	borate in boarding
vboreml	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	boreml in boarding
vborem2	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	borem2 in boarding
vborem3	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	borem3 in boarding
vbosrise	time	add_brd.prg add_sbd.prg mod_brd.prg	bsunrise in boarding
vbosset	time	add_brd.prg add_sbd.prg mod_brd.prg	bsunset in boarding
vbotime	time	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	boarding time
vbounit	text	add_brd.prg add_sbd.prg mod_brd.prg	bounit in boarding
vcall	text	add_brd.prg add_sar.prg add_sbd.prg	vessel call sign

Name	Type	Used In	Description
vcbcolor	text	add_brd.prg add_sar.prg add_sbd.prg	cabin color
vceiling	text	add_sar.prg add_sbd.prg mod_sar.prg	ceiling in sar
vchild	integer	add_brd.prg add_sbd.prg mod_brd.prg	childpfd in baording
vclouds	text	add_sar.prg add_sbd.prg	clouds in sar
vcmeyear	text	add_brd.prg add_sar.prg add_sbd.prg	cme decal year
vconstr	text	add_brd.prg add_sar.prg add_sbd.prg mod_vhin.prg	coded vessel construction
vdate	date	quer_per.prg quer_vsl.prg	intelligence date
vdateos	date	add_sar.prg add_sbd.prg add_sut.prg mod_sut.prg	on scene date
vdecal	text	add_brd.prg add_sar.prg add_sbd.prg	cme decal?
vdistoff	integer	add_sar.prg add_sbd.prg mod_sar.prg quer_per.prg quer_vsl.prg	distance offshore
vdkcolor	text	add_brd.prg add_sar.prg add_sbd.prg	deck color
vdob	date	quer_vsl.prg	date of birth
vdvmoord	date	add_sar.prg add_sbd.prg add_sut.prg mod_sut.prg	date distressed vessel mo
vdvmoort	time	add_sar.prg add_sbd.prg add_sut.prg mod_sut.prg	time distressed vessel mo
vepic	text	am_vint.prg quer_per.prg quer_vsl.prg	epiccode in vslintel
vetadate	date	add_sar.prg add_sbd.prg add_sut.prg mod_sut.prg	on scene eta date
vfname	text	am_pint.prg quer_per.prg quer_vsl.prg	intfname in perintel
vfoldno	text	add_sar.prg add_sbd.prg add_sut.prg mod_sar.prg mod_sut.prg quer_per.prg quer_vsl.prg	sar folder number
vhlcolor	text	add_brd.prg add_sar.prg add_sbd.prg	hull color

Name	Type	Used In	Description
vhome	text	add_brd.prg add_sar.prg add_sbd.prg	vessel homeport
vimmed	text	add_sar.prg add_sbd.prg mod_sar.prg quer_per.prg quer_vsl.prg	immediate sar?
vintdate	date	am_pint.prg am_pint.prg am_vint.prg quer_per.prg quer_vsl.prg	pintdate in perintel
vintdob	date	am_pint.prg	intdob in perintel
vintmi	text	am_pint.prg	intmi in perintel
vintprn	text	quer_vsl.prg	user input
vintrem1	text	am_pint.prg am_vint.prg	pintrem1 in perintel
vintrem2	text	am_pint.prg am_vint.prg	pintrem2 in perintel
vintrem3	text	am_pint.prg am_vint.prg	pintrem3 in perintel
viol54s	integer	violfreq.prg	counting variable
viol55s	integer	violfreq.prg	counting variable
viol56s	integer	violfreq.prg	counting variable
viol57s	integer	violfreq.prg	counting variable
viol58s	integer	violfreq.prg	counting variable
viol59s	integer	violfreq.prg	counting variable
viol60s	integer	violfreq.prg	counting variable
viol61s	integer	violfreq.prg	counting variable
viol62s	integer	violfreq.prg	counting variable
viol63s	integer	violfreq.prg	counting variable
viol64s	integer	violfreq.prg	counting variable
viol65s	integer	violfreq.prg	counting variable
viol66s	integer	violfreq.prg	counting variable
viol67s	integer	violfreq.prg	counting variable
viol68s	integer	violfreq.prg	counting variable
vlname	text	am_pint.prg quer_per.prg quer_vsl.prg	intlname in perintel
vltndesc	text	intelrpt.prg	= violdesc in VIOLS
vmarks	text	add_brd.prg add_sar.prg add_sbd.prg	vessel unusual markings
vmasts	text	add_brd.prg add_sar.prg add_sbd.prg	number of vessel masts
vmaxop	integer	add_brd.prg add_sar.prg add_sbd.prg	comparison variable
vmaxoper	integer	add_sbd.prg	comparison variable
vmaxown	integer	add_brd.prg add_sar.prg add_sbd.prg	comparison variable
vmi	text	quer_per.prg quer_vsl.prg	middle initial
vmucn	text	add_sar.prg add_sbd.prg mod_sar.prg quer_per.prg quer_vsl.prg	multi-unit case number
vno	text	intelrpt.prg	= violno in VIOLS

Name	Type	Used In	Description
vobcount	text	add_brd.prg add_sbd.prg mod_brd.prg	obscount in boarding
vobsbody	text	add_brd.prg add_sbd.prg mod_brd.prg	obsbody in boarding
vobscity	text	add_brd.prg add_sbd.prg mod_brd.prg	obscity in boarding
vobslat	real	add_brd.prg add_sbd.prg mod_brd.prg	latitude position
vobslong	real	add_brd.prg add_sbd.prg mod_brd.prg	longitude position
vobstate	text	add_brd.prg add_sbd.prg mod_brd.prg	obsstate in boarding
voffno	text	add_brd.prg add_sar.prg add_sbd.prg mod_vhin.prg quer_per.prg quer_vsl.prg	vessel official numbers
vopaddr	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg quer_per.prg quer_vsl.prg	operator street address
vopcity	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg quer_per.prg quer_vsl.prg	operator city address
vopcrse	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg quer_per.prg quer_vsl.prg	operator course code
vopdob	date	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg quer_per.prg quer_vsl.prg	operator date of birth
vopfname	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg quer_per.prg quer_vsl.prg	operator first name
vopid	integer	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg	operator id (system)
voplname	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg quer_per.prg quer_vsl.prg	operator last name

Name	Type	Used In	Description
vopmi	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg quer_per.prg quer_vsl.prg	operator middle initial
vopst	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg quer_per.prg quer_vsl.prg	operator state address
voptelno	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg quer_per.prg quer_vsl.prg	operator telephone number
vopzip	text	add_brd.prg add_sar.prg add_sbd.prg mod_owop.prg quer_per.prg quer_vsl.prg	operator zip code
voseta	time	add_sar.prg add_sbd.prg add_sut.prg mod_sut.prg	on scene eta time
vostatus	text	add_brd.prg add_sbd.prg mod_brd.prg	owstatus in boarding
vostime	time	add_sar.prg add_sbd.prg add_sut.prg mod_sut.prg	on scene time
vowaddr	text	add_brd.prg add_ownr.prg add_sar.prg add_sbd.prg mod_owop.prg quer_vsl.prg	owner street address
vowcity	text	add_brd.prg add_ownr.prg add_sar.prg add_sbd.prg mod_owop.prg quer_vsl.prg	owner city address
vowdob	date	add_brd.prg add_ownr.prg add_sar.prg add_sbd.prg mod_owop.prg quer_vsl.prg	owner date of birth
vowfname	text	add_brd.prg add_ownr.prg add_sar.prg add_sbd.prg mod_owop.prg quer_vsl.prg	owner first name
vowid	integer	quer_per.prg	owner record id number
vowlname	text	add_brd.prg add_ownr.prg add_sar.prg add_sbd.prg mod_owop.prg quer_vsl.prg	owner last name

Name	Type	Used In	Description
vowmi	text	add_brd.prg add_ownr.prg add_sar.prg add_sbd.prg mod_owop.prg quer_vsl.prg	owner middle initial
vownid	text	add_brd.prg add_brd.prg add_ownr.prg add_sar.prg add_sbd.prg add_sbd.prg mod_owop.prg mod_owop.prg	owner id (system)
vowst	text	add_brd.prg add_ownr.prg add_sar.prg add_sbd.prg mod_owop.prg quer_vsl.prg	owner state address
vowtelno	text	add_brd.prg add_ownr.prg add_sar.prg add_sbd.prg mod_owop.prg quer_vsl.prg	owner telephone number
vowzip	text	add_brd.prg add_ownr.prg add_sar.prg add_sbd.prg mod_owop.prg quer_vsl.prg	owner zip code
vpct2030	real	intelrpt.prg	percentage
vpct3040	real	intelrpt.prg	percentage
vpct4050	real	intelrpt.prg	percentage
vpct5060	real	intelrpt.prg	percentage
vpct6070	real	intelrpt.prg	percentage
vpct70	real	cmerpt.prg	percentage
vpct7080	real	intelrpt.prg	percentage
vpct71	real	cmerpt.prg	percentage
vpct76	real	cmerpt.prg	percentage
vpct80	real	cmerpt.prg	percentage
vpct8090	real	intelrpt.prg	percentage
vpct82	real	cmerpt.prg	percentage
vpct84	real	cmerpt.prg	percentage
vpct86	real	cmerpt.prg	percentage
vpct88	real	cmerpt.prg	percentage
vpct90	real	cmerpt.prg	percentage
vpct9010	real	intelrpt.prg	percentage
vpct<00	real	intelrpt.prg	percentage
vpct<20	real	intelrpt.prg	percentage
vpct<60	real	intelrpt.prg	percentage
vpct<65	real	intelrpt.prg	percentage
vpct<70	real	intelrpt.prg	percentage
vpct<75	real	intelrpt.prg	percentage
vpct<80	real	intelrpt.prg	percentage
vpct<85	real	intelrpt.prg	percentage
vpct<90	real	intelrpt.prg	percentage
vpct<95	real	intelrpt.prg	percentage
vpct>00	real	intelrpt.prg	percentage
vpct>100	real	intelrpt.prg	percentage
vperaddr	text	quer_per.prg	street address
vpercity	text	quer_per.prg	city address
vperdob	date	quer_per.prg	date of birth
vpermi	text	quer_per.prg	middle initial

Name	Type	Used In	Description
vperst	text	quer_per.prg	state address
vpertel	text	quer_per.prg	telephone number
vperzip	text	quer_per.prg	zip code
vpfname	text	quer_per.prg	first name
vplname	text	quer_per.prg	last name
vrem1	text	quer_per.prg quer_vsl.prg	personnel intelligence re
vrem2	text	quer_per.prg quer_vsl.prg	personnel intelligence re
vrem3	text	quer_per.prg quer_vsl.prg	personnel intelligence re
vsailno	text	add_brd.prg add_sar.prg add_sbd.prg	sail number
vsardate	date	add_sar.prg add_sbd.prg mod_sar.prg quer_per.prg quer_vsl.prg	sar date
vsarday	text	add_sar.prg add_sbd.prg mod_sar.prg quer_per.prg quer_vsl.prg	sar day of the week
vsarhin	text	mod_sar.prg	sarvhin in sar
vsarlat	real	add_sar.prg add_sbd.prg mod_sar.prg quer_per.prg quer_vsl.prg	latitude position
vsarlong	real	add_sar.prg add_sbd.prg mod_sar.prg quer_per.prg quer_vsl.prg	longitude position
vsarnod	text	add_sar.prg add_sbd.prg mod_sar.prg	nature of distress
vsarpob	integer	add_sar.prg add_sbd.prg mod_sar.prg quer_per.prg quer_vsl.prg	personnel on board
vsartime	time	add_sar.prg add_sbd.prg mod_sar.prg quer_per.prg quer_vsl.prg	sar time
vseas	text	add_sar.prg add_sbd.prg mod_sar.prg	seas in sar
vslcolor	text	add_brd.prg add_sar.prg add_sbd.prg	sail color
vslfound	text	mod_vhin.prg	flag variable
vslprn	text	quer_per.prg quer_vsl.prg	user input
vstolen	text	am_vint.prg quer_per.prg quer_vsl.prg	stolen in vslintel
vsumoord	date	add_sar.prg add_sbd.prg add_sut.prg mod_sut.prg	date sar unit moored

Name	Type	Used In	Description
vsumoort	time	add_sar.prg add_sbd.prg add_sut.prg mod_sut.prg	time sar unit moored
vsunrise	time	add_sar.prg add_sbd.prg mod_sar.prg	bsunrise in boarding
vsunset	time	add_sar.prg add_sbd.prg mod_sar.prg	bsunset in boarding
vsuper	text	add_brd.prg add_sar.prg add_sbd.prg	vessel superstructure
vswells	text	add_sar.prg add_sbd.prg mod_sar.prg	swells in sar
vtot1	integer	violfreq.prg	summation variable
vtot10	integer	violfreq.prg	summation variable
vtot11	integer	violfreq.prg	summation variable
vtot12	integer	violfreq.prg	summation variable
vtot13	integer	violfreq.prg	summation variable
vtot2	integer	violfreq.prg	summation variable
vtot3	integer	violfreq.prg	summation variable
vtot4	integer	violfreq.prg	summation variable
vtot5	integer	violfreq.prg	summation variable
vtot6	integer	violfreq.prg	summation variable
vtot7	integer	violfreq.prg	summation variable
vtot8	integer	violfreq.prg	summation variable
vtot9	integer	violfreq.prg	summation variable
vtotal	integer	violfreq.prg	total violations
vtowdate	date	add_sar.prg add_sbd.prg add_sut.prg mod_sut.prg	date started towing
vtowtime	time	add_sar.prg add_sbd.prg add_sut.prg mod_sut.prg	time started towing
vtrcolor	text	add_brd.prg add_sar.prg add_sbd.prg	trim color
vucn	text	add_sar.prg add_sbd.prg mod_sar.prg quer_per.prg quer_vsl.prg	unit case number
vunit	text	add_sar.prg add_sbd.prg add_sut.prg mod_sut.prg	sar unit name
vuns69	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	unsafe condition 69
vuns70	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	unsafe condition 70
vuns71	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	unsafe condition 71

Name	Type	Used In	Description
vuns72	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	unsafe condition 72
vuns73	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	unsafe condition 73
vuns74	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	unsafe condition 74
vuns75	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	unsafe condition 75
vuns76	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	unsafe condition 76
vuns77	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	unsafe condition 77
vuns78	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	unsafe condition 78
vuns79	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	unsafe condition 79
vuwdate	date	add_sar.prg add_sbd.prg add_sut.prg mod_sut.prg	underway date
vuwtime	time	add_sar.prg add_sbd.prg add_sut.prg mod_sut.prg	underway time
vvalue	dollar	add_brd.prg add_sar.prg add_sbd.prg	vessel value
vviol54	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	violation code 54
vviol55	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	violation code 55

Name	Type	Used In	Description
vviol56	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	violation code 56
vviol57	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	violation code 57
vviol58	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	violation code 58
vviol59	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	violation code 59
vviol60	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	violation code 60
vviol61	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	violation code 61
vviol62	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	violation code 62
vviol63	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	violation code 63
vviol64	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	violation code 64
vviol65	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	violation code 65
vviol66	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	violation code 66
vviol67	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	violation code 67
vviol68	text	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	violation code 68

Name	Type	Used In	Description
vvisdist	integer	add_sar.prg add_sbd.prg mod_sar.prg	visdist in sar
vvrem1	text	quer_per.prg quer_vsl.prg	vessel intelligence remar
vvrem2	text	quer_per.prg quer_vsl.prg	vessel intelligence remar
vvrem3	text	quer_per.prg quer_vsl.prg	vessel intelligence remar
vvsleng	text	add_brd.prg add_sar.prg add_sbd.prg mod_vhin.prg	coded vessel engine compa
vvslft	integer	add_brd.prg add_sar.prg add_sbd.prg mod_vhin.prg	vessel length (feet)
vvslfuel	text	add_brd.prg add_sar.prg add_sbd.prg mod_vhin.prg	coded fuel compartment
vvslhin	text	add_brd.prg add_ownr.prg add_sar.prg add_sbd.prg mod_brd.prg mod_vhin.prg quer_per.prg quer_vsl.prg	vslhin in vessel
vvslhp	text	add_brd.prg add_sar.prg add_sbd.prg mod_vhin.prg	vessel horsepower
vvslhull	text	add_brd.prg add_sar.prg add_sbd.prg mod_vhin.prg	coded vessel hull materia
vvslin	integer	add_brd.prg add_sar.prg add_sbd.prg mod_vhin.prg	vessel length (inches)
vvslmake	text	add_brd.prg add_sar.prg add_sbd.prg mod_vhin.prg quer_per.prg quer_vsl.prg	vessel make
vvslmod	text	add_brd.prg add_sar.prg add_sbd.prg mod_vhin.prg	vslmodel in vessel
vvslname	text	add_brd.prg add_sar.prg add_sbd.prg am_vint.prg mod_vhin.prg quer_per.prg quer_vsl.prg	vessel name
vvslno	text	add_brd.prg add_sar.prg add_sbd.prg mod_vhin.prg quer_per.prg quer_vsl.prg	vessel number

Name	Type	Used In	Description
vvslpob	integer	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	personnel on board
vvslprop	text	add_brd.prg add_sar.prg add_sbd.prg mod_vhin.prg	coded vessel propulsion
vvslton	integer	add_brd.prg add_sar.prg add_sbd.prg mod_vhin.prg	vsltons in vessel
vvsltype	text	add_sar.prg add_sbd.prg add_brd.prg mod_vhin.prg	coded vessel type
vvsluse	text	add_brd.prg add_sar.prg add_sbd.prg mod_vhin.prg	coded vessel use
vvslyear	text	add_brd.prg add_sar.prg add_sbd.prg mod_vhin.prg	vessel year
vweapons	integer	add_brd.prg add_sbd.prg mod_brd.prg quer_per.prg quer_vsl.prg	weapons in boarding
vwinddir	text	add_sar.prg add_sbd.prg mod_sar.prg	winddir in sar
vwindspd	text	add_sar.prg add_sbd.prg mod_sar.prg	windspd in sar
wedsday	integer	sarday.prg	counting variable
westtst	real	bomajor.prg bominor.prg sarmajor.prg sarminor.prg	nautical position coord.
year	integer	sarday.prg	saryear
yr1981	date	sarday.prg	starting date
yr1981a	date	sarday.prg	ending date
yr1982	date	sarday.prg	starting date
yr1982a	date	sarday.prg	ending date
yr1983	date	sarday.prg	starting date
yr1983a	date	sarday.prg	ending date
yr1984	date	sarday.prg	starting date
yr1984a	date	sarday.prg	ending date
yr1985	date	sarday.prg	starting date
yr1985a	date	sarday.prg	ending date
yr1986	date	sarday.prg	starting date
yr1986a	date	sarday.prg	ending date
yr1987	date	sarday.prg	starting date
yr1987a	date	sarday.prg	ending date
yr1988	date	sarday.prg	starting date
yr1988a	date	sarday.prg	ending date
yr1989	date	sarday.prg	starting date
yr1989a	date	sarday.prg	ending date
yr1990	date	sarday.prg	starting date
yr1990a	date	sarday.prg	ending date
yr1991	date	sarday.prg	starting date
yr1991a	date	sarday.prg	ending date
yr1992	date	sarday.prg	starting date

Name	Type	Used In	Description
yr1992a	date	sarday.prg	ending date
yr1993	date	sarday.prg	starting date
yr1993a	date	sarday.prg	ending date
yr1994	date	sarday.prg	starting date
yr1994a	date	sarday.prg	ending date
yr1995	date	sarday.prg	starting date
yr1995a	date	sarday.prg	ending date
yr1996	date	sarday.prg	starting date
yr1996a	date	sarday.prg	ending date
yr1997	date	sarday.prg	starting date
yr1997a	date	sarday.prg	ending date
yr1998	date	sarday.prg	starting date
yr1998a	date	sarday.prg	ending date
yr1999	date	sarday.prg	starting date
yr1999a	date	sarday.prg	ending date
yr2000	date	sarday.prg	starting date
yr2000a	date	sarday.prg	ending date
yr2001	date	sarday.prg	starting date
yr2001a	date	sarday.prg	ending date
yr2002	date	sarday.prg	starting date
yr2002a	date	sarday.prg	ending date
yr2003	date	sarday.prg	starting date
yr2003a	date	sarday.prg	ending date
yr2004	date	sarday.prg	starting date
yr2004a	date	sarday.prg	ending date
yr2005	date	sarday.prg	starting date
yr2005a	date	sarday.prg	ending date
yr2006	date	sarday.prg	starting date
yr2006a	date	sarday.prg	ending date
yr2007	date	sarday.prg	starting date
yr2007a	date	sarday.prg	ending date
yr2008	date	sarday.prg	starting date
yr2008a	date	sarday.prg	ending date
yr2009	date	sarday.prg	starting date
yr2009a	date	sarday.prg	ending date
yr2010	date	sarday.prg	starting date
yr2010a	date	sarday.prg	ending date
yr2011	date	sarday.prg	starting date
yr2011a	date	sarday.prg	ending date
yr2012	date	sarday.prg	starting date
yr2012a	date	sarday.prg	ending date
yr71to75	integer	cmerpt.prg	counting variable
yr76to79	integer	cmerpt.prg	counting variable
yr80to81	integer	cmerpt.prg	counting variable
yr82to83	integer	cmerpt.prg	counting variable
yr84to85	integer	cmerpt.prg	counting variable
yr86to87	integer	cmerpt.prg	counting variable
yr88to89	integer	cmerpt.prg	counting variable
yr<1960	integer	intelrpt.prg	counting variable
yr<1965	integer	intelrpt.prg	counting variable
yr<1970	integer	intelrpt.prg	counting variable
yr<1975	integer	intelrpt.prg	counting variable
yr<1980	integer	intelrpt.prg	counting variable
yr<1985	integer	intelrpt.prg	counting variable
yr<1990	integer	intelrpt.prg	counting variable
yr<1995	integer	intelrpt.prg	counting variable
yr<2000	integer	intelrpt.prg	counting variable
yr>2000	integer	intelrpt.prg	counting variable
yrtest1	integer	sarday.prg	leap year test
yrtest2	integer	sarday.prg	leap year test

Program Variables Sorted by Where Used

Used In	Name	Type	Description
add_brd.prg	b1	date	bodate in boarding
	b2	time	botime in boarding
	b3	real	obslat in boarding
	b4	real	bolong in boarding
	backup	text	flag variable
	choice	text	user input variable
	cont	text	user input
	difbo	text	flag variable
	fixopr	text	flag variable
	fixown	text	flag variable
	loadmenu	text	flag variable
	more	text	flag variable
	mropers	text	user input
	mrownrs	text	user input
	op1	text	oplname in operator
	op10	text	opcourse in operator
	op11	integer	opid in operator
	op2	text	opfname in operator
	op3	text	opmi in operator
	op4	text	opaddr in operator
	op5	text	opcity in operator
	op6	text	opstate in operator
	op7	text	opzip in operator
	op8	text	optelno in operator
	op9	date	opdob in operator
	ow1	text	ownlname in owners
	ow10	text	ownid in owners
	ow2	text	ownfname in owners
	ow3	text	ownmi in owners
	ow4	text	ownaddr in owners
	ow5	text	owncity in owners
	ow6	text	ownstate in owners
	ow7	text	ownzip in owners
	ow8	text	owntelno in owners
	ow9	date	owndob in owners
	pg	integer	counting variable
	pgcount	integer	counting variable
	review	text	flag variable
	v1	text	vslno in vessel
	v10	text	vslhp in vessel
	v11	text	vsluse in vessel
	v12	text	vsltype in vessel
	v13	text	vslprop in vessel
	v14	text	vslhull in vessel
	v15	text	vsleng in vessel
	v16	text	vslfuel in vessel
	v17	text	vslconst in vessel
	v18	text	cmedecal in vessel
	v19	text	cmeyuear in vessel
	v2	text	vslname in vessel
	v20	text	vsloffno in vessel
	v21	text	homeport in vessel
	v22	text	vslcall in vessel
	v23	text	vslvalue in vessel
	v24	text	vslsuper in vessel
	v25	text	vslmasts in vessel
	v26	text	sailno in vessel
	v27	text	hlcolor in vessel
	v28	text	trcolor in vessel
	v29	text	slcolor in vessel
	v3	text	vslhin in vessel
	v30	text	cbcolor in vessel
	v31	text	dkcolor in vessel
	v32	text	vslmarks in vessel
	v4	text	vslmake in vessel

Used In	Name	Type	Description
add_brd.prg	v5	text	vslmodel in vessel
	v6	text	vslyear in vessel
	v7	integer	vsltons in vessel
	v8	integer	vsllenft in vessel
	v9	integer	vsllenin in vessel
	vadult	integer	adultpfd in boarding
	vbodate	date	boarding date
	vboday	text	boday in boarding
	vbodist	text	bodist in boarding
	vboname	text	boname in boarding
	vbono	text	boarding report number
	vboopfac	text	boopfac in boarding
	vborate	text	borate in boarding
	vborem1	text	borem1 in boarding
	vborem2	text	borem2 in boarding
	vborem3	text	borem3 in boarding
	vbosrise	time	bsunrise in boarding
	vbosset	time	bsunset in boarding
	vbotime	time	boarding time
	vbounit	text	bounit in boarding
	vcall	text	vessel call sign
	vcbcolor	text	cabin color
	vchild	integer	childpfd in baording
	vcmeyear	text	cme decal year
	vconstr	text	coded vessel construction
	vdecal	text	cme decal?
	vdkcolor	text	deck color
	vhlcolor	text	hull color
	vhome	text	vessel homeport
	vmarks	text	vessel unusual markings
	vmasts	text	number of vessel masts
	vmaxop	integer	comparison variable
	vmaxown	integer	comparison variable
	vobcount	text	obscount in boarding
	vobsbody	text	obsbody in boarding
	vobscity	text	obscity in boarding
	vobslat	real	latitude position
	vobslong	real	longitude position
	vobstate	text	obsstate in boarding
	voffno	text	vessel official numbers
	vopaddr	text	operator street address
	vopcity	text	operator city address
	vopcrse	text	operator course code
	vopdob	date	operator date of birth
	vopfname	text	operator first name
	vopid	integer	operator id (system)
	voplname	text	operator last name
	vopmi	text	operator middle initial
	vopst	text	operator state address
	voptelno	text	operator telephone number
	vopzip	text	operator zip code
	vostatus	text	owstatus in boarding
	vowaddr	text	owner street address
	vowcity	text	owner city address
	vowdob	date	owner date of birth
	vowfname	text	owner first name
	vowlname	text	owner last name
	vowmi	text	owner middle initial
	vownid	text	owner id (system)

Used In	Name	Type	Description
add_brd.prg	vownid	integer	owner id (system)
	vowst	text	owner state address
	vowtelno	text	owner telephone number
	vowzip	text	owner zip code
	vsailno	text	sail number
	vslcolor	text	sail color
	vsuper	text	vessel superstructure
	vtrcolor	text	trim color
	vuns69	text	unsafe condition 69
	vuns70	text	unsafe condition 70
	vuns71	text	unsafe condition 71
	vuns72	text	unsafe condition 72
	vuns73	text	unsafe condition 73
	vuns74	text	unsafe condition 74
	vuns75	text	unsafe condition 75
	vuns76	text	unsafe condition 76
	vuns77	text	unsafe condition 77
	vuns78	text	unsafe condition 78
	vuns79	text	unsafe condition 79
	vvalue	dollar	vessel value
	vviol54	text	violation code 54
	vviol55	text	violation code 55
	vviol56	text	violation code 56
	vviol57	text	violation code 57
	vviol58	text	violation code 58
	vviol59	text	violation code 59
	vviol60	text	violation code 60
	vviol61	text	violation code 61
	vviol62	text	violation code 62
	vviol63	text	violation code 63
	vviol64	text	violation code 64
	vviol65	text	violation code 65
	vviol66	text	violation code 66
	vviol67	text	violation code 67
	vviol68	text	violation code 68
	vvsleng	text	coded vessel engine compa
	vvslft	integer	vessel length (feet)
	vvslfuel	text	coded fuel compartment
	vvslhin	text	vslhin in vessel
	vvslhp	text	vessel horsepower
	vvslhull	text	coded vessel hull materia
	vvslin	integer	vessel length (inches)
	vvslmake	text	vessel make
	vvslmod	text	vslmodel in vessel
	vvslname	text	vessel name
	vvslno	text	vessel number
	vvslpob	integer	personnel on board
	vvslprop	text	coded vessel propulsion
	vvslton	integer	vsltons in vessel
	vvsltype	text	coded vessel type
	vvsluse	text	coded vessel use
	vvslyear	text	vessel year
	vweapons	integer	weapons in boarding
add_ownr.prg	done	text	flag variable
	loadok	text	flag variable
	more	text	flag variable
	mrownrs	text	user input
	out	text	flag variable
	reenter	text	user input
	vowaddr	text	owner street address
	vowcity	text	owner city address
	vowdob	date	owner date of birth
	vowfname	text	owner first name
	vowlname	text	owner last name
	vowmi	text	owner middle initial

Used In	Name	Type	Description
add_ownr.prg	vownid	integer	owner id (system)
	vowst	text	owner state address
	vowtelno	text	owner telephone number
	vowzip	text	owner zip code
	vvslhin	text	vslhin in vessel
add_sar.prg	addlownr	text	user input
	backup	text	flag variable
	choice	text	user input variable
	cont	text	user input
	difsar	text	flag variable
	fixopr	text	flag variable
	fixown	text	flag variable
	loadmenu	text	flag variable
	more	text	flag variable
	mrownrs	text	user input
	mrunits	text	user input
	op1	text	oplname in operator
	op10	text	opcourse in operator
	op11	integer	opid in operator
	op2	text	opfname in operator
	op3	text	opmi in operator
	op4	text	opaddr in operator
	op5	text	opcity in operator
	op6	text	opstate in operator
	op7	text	opzip in operator
	op8	text	optelno in operator
	op9	date	opdob in operator
	ow1	text	ownlname in owners
	ow10	text	ownid in owners
	ow2	text	ownfname in owners
	ow3	text	ownmi in owners
	ow4	text	ownaddr in owners
	ow5	text	owncity in owners
	ow6	text	ownstate in owners
	ow7	text	ownzip in owners
	ow8	text	owntelno in owners
	ow9	date	owndob in owners
	pg	integer	counting variable
	pgcount	integer	counting variable
	review	text	flag variable
	s1	date	sardate in sar
	s2	time	sartime in sar
	s3	real	sarlat in sar
	s4	real	sarlong in sar
	s5	text	sarnod in sar
	s6	text	sarvhin in sar
	sunits	text	user input
	v1	text	vslno in vessel
	v10	text	vslhp in vessel
	v11	text	vsluse in vessel
	v12	text	vsltype in vessel
	v13	text	vslprop in vessel
	v14	text	vslhull in vessel
	v15	text	vsleng in vessel
	v16	text	vslfuel in vessel
	v17	text	vslconst in vessel
	v18	text	cmedecal in vessel
	v19	text	cmeyuear in vessel
	v2	text	vslname in vessel
	v20	text	vsloffno in vessel
	v21	text	homeport in vessel
	v22	text	vslcall in vessel
	v23	text	vslvalue in vessel
	v24	text	vslsuper in vessel
	v25	text	vslmasts in vessel

Used In	Name	Type	Description
add_sar.prg	v26	text	sailno in vessel
	v27	text	hlcolor in vessel
	v28	text	trcolor in vessel
	v29	text	slcolor in vessel
	v3	text	vslhin in vessel
	v30	text	cbcolor in vessel
	v31	text	dkcolor in vessel
	v32	text	vslmarks in vessel
	v4	text	vslmake in vessel
	v5	text	vslmodel in vessel
	v6	text	vslyear in vessel
	v7	integer	vsltons in vessel
	v8	integer	vsllenft in vessel
	v9	integer	vsllenin in vessel
	valgdate	date	date alongside
	valgtime	time	time alongside
	vcall	text	vessel call sign
	vcbcolor	text	cabin color
	vceiling	text	ceiling in sar
	vclouds	text	clouds in sar
	vcmeyear	text	cme decal year
	vconstr	text	coded vessel construction
	vdateos	date	on scene date
	vdecal	text	cme decal?
	vdistoff	integer	distance offshore
	vdkcolor	text	deck color
	vdvmoord	date	date distressed vessel mo
	vdvmoort	time	time distressed vessel mo
	vetadate	date	on scene eta date
	vfoldno	text	sar folder number
	vhlcolor	text	hull color
	vhome	text	vessel homeport
	vimmed	text	immediate sar?
	vmarks	text	vessel unusual markings
	vmasts	text	number of vessel masts
	vmaxop	integer	comparison variable
	vmaxown	integer	comparison variable
	vmucn	text	multi-unit case number
	voffno	text	vessel official numbers
	vopaddr	text	operator street address
	vopcity	text	operator city address
	vopcrse	text	operator course code
	vopdob	date	operator date of birth
	vopfname	text	operator first name
	vopid	integer	operator id (system)
	voplname	text	operator last name
	vopmi	text	operator middle initial
	vopst	text	operator state address
	voptelno	text	operator telephone number
	vopzip	text	operator zip code
	voseta	time	on scene eta time
	vostime	time	on scene time
	vowaddr	text	owner street address
	vowcity	text	owner city address
	vowdob	date	owner date of birth
	vowfname	text	owner first name
	vowlname	text	owner last name
	vowmi	text	owner middle initial
	vownid	text	owner id (system)
	vowst	text	owner state address
	vowtelno	text	owner telephone number
	vowzip	text	owner zip code
	vsailno	text	sail number
	vsardate	date	sar date
	vsarday	text	sar day of the week

Used In	Name	Type	Description
add_sar.prg	vsarlat	real	latitude position
	vsarlong	real	longitude position
	vsarnod	text	nature of distress
	vsarpob	integer	personnel on board
	vsartime	time	sar time
	vseas	text	seas in sar
	vslcolor	text	sail color
	vvsltype	text	coded vessel type
	vsumoord	date	date sar unit moored
	vsumoort	time	time sar unit moored
	vsunrise	time	bsunrise in boarding
	vsunset	time	bsunset in boarding
	vsuper	text	vessel superstructure
	vswells	text	swells in sar
	vtowdate	date	date started towing
	vtowtime	time	time started towing
	vtrcolor	text	trim color
	vucn	text	unit case number
	vunit	text	sar unit name
	vuwdate	date	underway date
	vuwtime	time	underway time
	vvalue	dollar	vessel value
	vvisdist	integer	visdist in sar
	vvsleng	text	coded vessel engine compa
	vvslft	integer	vessel length (feet)
	vvslfuel	text	coded fuel compartment
	vvslhin	text	vslhin in vessel
	vvslhp	text	vessel horsepower
	vvslhull	text	coded vessel hull materia
	vvslin	integer	vessel length (inches)
	vvslmake	text	vessel make
	vvslmod	text	vslmodel in vessel
	vvslname	text	vessel name
	vvslno	text	vessel number
	vvslprop	text	coded vessel propulsion
	vvslton	integer	vsltons in vessel
	vvsluse	text	coded vessel use
	vvslyear	text	vessel year
	vwinddir	text	winddir in sar
	vwindspd	text	windspd in sar
add_sbd.prg	addl_opr	text	user input
	addl_own	text	user input
	b1	date	bodate in boarding
	b2	time	botime in boarding
	b3	real	obslat in boarding
	b4	real	bolong in boarding
	backup	text	flag variable
	choice	text	user input variable
	cont	text	user input
	difbo	text	flag variable
	difsar	text	flag variable
	fixopr	text	flag variable
	fixown	text	flag variable
	loadmenu	text	flag variable
	more	text	flag variable
	mropers	text	user input
	mrownrs	text	user input
	mrunits	text	user input
	op1	text	oplname in operator
	op10	text	opcourse in operator
	op11	integer	opid in operator
	op2	text	opfname in operator
	op3	text	opmi in operator
	op4	text	opaddr in operator
	op5	text	opcity in operator

Used In	Name	Type	Description
add_sbd.prg	op6	text	opstate in operator
	op7	text	opzip in operator
	op8	text	optelno in operator
	op9	date	opdob in operator
	ow1	text	ownlname in owners
	ow10	text	ownid in owners
	ow2	text	ownfname in owners
	ow3	text	ownmi in owners
	ow4	text	ownaddr in owners
	ow5	text	owncity in owners
	ow6	text	ownstate in owners
	ow7	text	ownzip in owners
	ow8	text	owntelno in owners
	ow9	date	owndob in owners
	pg	integer	counting variable
	pgcount	integer	counting variable
	review	text	flag variable
	s1	date	sardate in sar
	s2	time	sartime in sar
	s3	real	sarlat in sar
	s4	real	sarlong in sar
	s5	text	sarnod in sar
	s6	text	sarvhin in sar
	sunits	text	user input
	v1	text	vslno in vessel
	v10	text	vslhp in vessel
	v11	text	vsluse in vessel
	v12	text	vsltype in vessel
	v13	text	vslprop in vessel
	v14	text	vslhull in vessel
	v15	text	vsleng in vessel
	v16	text	vslfuel in vessel
	v17	text	vslconst in vessel
	v18	text	cmedecal in vessel
	v19	text	cmeyuear in vessel
	v2	text	vslname in vessel
	v20	text	vsloffno in vessel
	v21	text	homeport in vessel
	v22	text	vslcall in vessel
	v23	text	vslvalue in vessel
	v24	text	vslsuper in vessel
	v25	text	vslmasts in vessel
	v26	text	sailno in vessel
	v27	text	hlcolor in vessel
	v28	text	trcolor in vessel
	v29	text	slcolor in vessel
	v3	text	vslhin in vessel
	v30	text	cbcolor in vessel
	v31	text	dkcolor in vessel
	v32	text	vslmarks in vessel
	v4	text	vslmake in vessel
	v5	text	vslmodel in vessel
	v6	text	vslyear in vessel
	v7	integer	vsltons in vessel
	v8	integer	vsllenft in vessel
	v9	integer	vsllenin in vessel
	vadult	integer	adultpfd in boarding
	valgdate	date	date alongside
	valgtime	time	time alongside
	vbodate	date	boarding date
	vboday	text	boday in boarding
	vbodist	text	bodist in boarding
	vboname	text	boname in boarding
	vbono	text	boarding repcrt number
	vboopfac	text	boopfac in boarding

Used In	Name	Type	Description
add_sbd.prg	vborate	text	borate in boarding
	vboreml	text	boreml in boarding
	vborem2	text	borem2 in boarding
	vborem3	text	borem3 in boarding
	vbosrise	time	bsunrise in boarding
	vbosset	time	bsunset in boarding
	vbotime	time	boarding time
	vbounit	text	bounit in boarding
	vcall	text	vessel call sign
	vcbcolor	text	cabin color
	vceiling	text	ceiling in sar
	vchild	integer	childpfd in baording
	vclouds	text	clouds in sar
	vcmeyear	text	cme decal year
	vconstr	text	coded vessel construction
	vdateos	date	on scene date
	vdecal	text	cme decal?
	vdistoff	integer	distance offshore
	vdkcolor	text	deck color
	vdvmoord	date	date distressed vessel mo
	vdvmoort	time	time distressed vessel mo
	vetadate	date	on scene eta date
	vfoldno	text	sar folder number
	vhlcolor	text	hull color
	vhome	text	vessel homeport
	vimmed	text	immediate sar?
	vmarks	text	vessel unusual markings
	vmasts	text	number of vessel masts
	vmaxop	integer	comparison variable
	vmaxoper	integer	comparison variable
	vmaxown	integer	comparison variable
	vmucn	text	multi-unit case number
	vobcount	text	obscount in boarding
	vobsbody	text	obsbody in boarding
	vobscity	text	obscity in boarding
	vobslat	real	latitude position
	vobslong	real	longitude position
	vobstate	text	obsstate in boarding
	voffno	text	vessel official numbers
	vopaddr	text	operator street address
	vopcity	text	operator city address
	vopcrse	text	operator course code
	vopdob	date	operator date of birth
	vopfname	text	operator first name
	vopid	integer	operator id (system)
	voplname	text	operator last name
	vopmi	text	operator middle initial
	vopst	text	operator state address
	voptelno	text	operator telephone number
	vopzip	text	operator zip code
	voseta	time	on scene eta time
	vostatus	text	owstatus in boarding
	vostime	time	on scene time
	vowaddr	text	owner street address
	vowcity	text	owner city address
	vowdob	date	owner date of birth
	vowfname	text	owner first name
	vowlname	text	owner last name
	vowmi	text	owner middle initial
	vownid	text	owner id (system)
	vownid	integer	owner id (system)
	vowst	text	owner state address
	vowtelno	text	owner telephone number
	vowzip	text	owner zip code
	vsailno	text	sail number

Used In	Name	Type	Description
add_sbd.prg	vsardate	date	sar date
	vsarday	text	sar day of the week
	vsarlat	real	latitude position
	vsarlong	real	longitude position
	vsarnod	text	nature of distress
	vsarpob	integer	personnel on board
	vsartime	time	sar time
	vseas	text	seas in sar
	vslcolor	text	sail color
	vvsltype	text	coded vessel type
	vsumoord	date	date sar unit moored
	vsumoort	time	time sar unit moored
	vsunrise	time	bsunrise in boarding
	vsunset	time	bsunset in boarding
	vsuper	text	vessel superstructure
	vswells	text	swells in sar
	vtowdate	date	date started towing
	vtowtime	time	time started towing
	vtrcolor	text	trim color
	vucn	text	unit case number
	vunit	text	sar unit name
	vuns69	text	unsafe condition 69
	vuns70	text	unsafe condition 70
	vuns71	text	unsafe condition 71
	vuns72	text	unsafe condition 72
	vuns73	text	unsafe condition 73
	vuns74	text	unsafe condition 74
	vuns75	text	unsafe condition 75
	vuns76	text	unsafe condition 76
	vuns77	text	unsafe condition 77
	vuns78	text	unsafe condition 78
	vuns79	text	unsafe condition 79
	vuwdate	date	underway date
	vuwtime	time	underway time
	vvalue	dollar	vessel value
	vviol54	text	violation code 54
	vviol55	text	violation code 55
	vviol56	text	violation code 56
	vviol57	text	violation code 57
	vviol58	text	violation code 58
	vviol59	text	violation code 59
	vviol60	text	violation code 60
	vviol61	text	violation code 61
	vviol62	text	violation code 62
	vviol63	text	violation code 63
	vviol64	text	violation code 64
	vviol65	text	violation code 65
	vviol66	text	violation code 66
	vviol67	text	violation code 67
	vviol68	text	violation code 68
	vvisdist	integer	visdist in sar
	vvsleng	text	coded vessel engine compa
	vvslft	integer	vessel length (feet)
	vvslfuel	text	coded fuel compartment
	vvslhin	text	vslhin in vessel
	vvslhp	text	vessel horsepower
	vvslhull	text	coded vessel hull materia
	vvslin	integer	vessel length (inches)
	vvslmake	text	vessel make
	vvslmod	text	vslmodel in vessel
	vvslname	text	vessel name
	vvslno	text	vessel number
	vvslpob	integer	personnel on board
	vvslprop	text	coded vessel propulsion
	vvslton	integer	vsltons in vessel

Used In	Name	Type	Description
add_sbd.prg	vvsluse	text	coded vessel use
	vvslyear	text	vessel year
	vweapons	integer	weapons in boarding
	vwinddir	text	winddir in sar
	vwindspd	text	windspd in sar
add_sut.prg	done	text	flag variable
	mrunits	text	user input
	valgdate	date	date alongside
	valgtime	time	time alongside
	vdateos	date	on scene date
	vdvmoord	date	date distressed vessel mo
	vdvmoort	time	time distressed vessel mo
	vetadate	date	on scene eta date
	vfoldno	text	sar folder number
	voseta	time	on scene eta time
	vostime	time	on scene time
	vsumoord	date	date sar unit moored
	vsumoort	time	time sar unit moored
	vtowdate	date	date started towing
	vtowtime	time	time started towing
	vunit	text	sar unit name
	vuwdate	date	underway date
	vuwtime	time	underway time
am_pint.prg	choice	text	user input
	correct	text	user input
	more	text	flag variable
	mrintel	text	user input
	reenter	text	user input
	vfname	text	intfname in perintel
	vintdate	date	pintdate in perintel
	vintdate	date	intdate in vslintel
	vintdob	date	intdob in perintel
	vintmi	text	intmi in perintel
	vintrem1	text	pintrem1 in perintel
	vintrem2	text	pintrem2 in perintel
	vintrem3	text	pintrem3 in perintel
	vlname	text	intlname in perintel
am_vint.prg	choice	text	user input
	correct	text	user input
	fixvsl	text	user input
	more	text	flag variable
	mrintel	text	user input
	reenter	text	user input
	vepic	text	epiccode in vslintel
	vintdate	date	intdate in vslintel
	vintrem1	text	vremark1 in vslintel
	vintrem2	text	vremark2 in vslintel
	vintrem3	text	vremark3 in vslintel
	vstolen	text	stolen in vslintel
	vvslname	text	vessel name
boarding.prg	datediff	integer	date difference
	enddate	date	ending date
	numrows	integer	counting variable
	strtdate	date	starting date
bomajor.prg	datdiff	integer	date difference
	easttst	real	nautical position coord.
	enddate	date	ending date
	inarea	integer	summation variable
	inpct	real	percentage
	northtst	real	nautical position coord.
	outarea	integer	summation variable
	outpct	real	percentage
	southtst	real	nautical position coord.
	strtdate	date	starting date
	temppct	real	percentage
	westtst	real	nautical position coord.

Used In	Name	Type	Description
bominor.prg	datadiff	integer	date difference
	datediff	integer	date difference
	easttst	real	nautical position coord.
	enddate	date	ending date
	inarea	integer	counting variable
	inpct	real	percentage
	northtst	real	nautical position coord.
	outarea	integer	counting variable
	outpct	real	percentage
	southtst	real	nautical position coord.
	strtdate	date	starting date
	temppct	real	percentage
	westtst	real	nautical position coord.
brd_main.prg	pcik6	integer	counting variable
cmerpt.prg	after90	integer	counting variable
	before70	integer	counting variable
	cmecount	integer	counting variable
	datediff	integer	date difference
	enddate	date	ending date
	pct7175	real	percentage
	pct7679	real	percentage
	pct8081	real	percentage
	pct8283	real	percentage
	pct8485	real	percentage
	pct8687	real	percentage
	pct8889	real	percentage
	pctcme	real	percentage
	pctgt90	real	percentage
	pctlt70	real	percentage
	strtdate	date	starting date
	totbrds	integer	counting variable
	vpct70	real	percentage
	vpct71	real	percentage
	vpct76	real	percentage
	vpct80	real	percentage
	vpct82	real	percentage
	vpct84	real	percentage
	vpct86	real	percentage
	vpct88	real	percentage
	vpct90	real	percentage
	yr71to75	integer	counting variable
	yr76to79	integer	counting variable
	yr80to81	integer	counting variable
	yr82to83	integer	counting variable
	yr84to85	integer	counting variable
	yr86to87	integer	counting variable
	yr88to89	integer	counting variable
daynight.prg	datediff	integer	date difference
	daybrd	integer	counting variable
	daysar	integer	counting variable
	enddate	date	ending date
	nitebrd	integer	counting variable
	nitesar	integer	counting variable
	pctdayb	real	percentage
	pctdays	real	percentage
	pctniteb	real	percentage
	pctnites	real	percentage
	strtdate	date	starting date
	totbrds	integer	total boardings
	totsars	integer	total sars

Used In	Name	Type	Description
epicdump.prg	datediff	integer	date difference
	edscl	text	= epicdscl in VSLINTEL
	edsc2	text	= epicdsc2 in VSLINTEL
	edsc3	text	= epicdsc3 in VSLINTEL
	edsc4	text	= epicdsc4 in VSLINTEL
	enddate	date	ending date
	epicnum	integer	counting variable
	epictest	text	= epiccode in VSLINTEL
	pcttemp1	real	% epiccode / total vsls
	pcttemp2	real	% epiccode / total vsls
	strtdate	date	starting date
	totepics	integer	counting variable
immedsar.prg	datediff	integer	date difference
	enddate	date	ending date
	notimmed	integer	non-immediate SAR Cases
	numrows	integer	counting variable
	pctimmed	real	% immediate SAR cases
	pctimmed	real	percentage
	pctnotim	real	% non-immediate SAR Cases
	pctnotim	real	percentage
	strtdate	date	starting date
	totimmed	integer	counting variable
	totsars	integer	counting variable
inpmain.prg	pick2	integer	menu variable
int_main.prg	pick5	integer	menu variable
intelrpt.prg	code	integer	input variable
	constrds	text	= consdesc in CONSTR
	consttst	text	= constype in CONSTR
	datediff	integer	date difference
	enddate	date	ending date
	engdescr	text	= ecomptyp in ENGCOMP
	engtst	text	= ecomptype in ENGCOMP
	fdesc	text	= fcompdes in FUELCOMP
	fueltst	text	= fcomptyp in FUELCOMP
	hdesc	text	= hmatdesc in HULLMAT
	hulltst	text	= hmattype in HULLMAT
	len2030	integer	counting variable
	len3040	integer	counting variable
	len4050	integer	counting variable
	len5060	integer	counting variable
	len6070	integer	counting variable
	len7080	integer	counting variable
	len8090	integer	counting variable
	len90100	integer	counting variable
	len<20	integer	counting variable
	len>100	integer	counting variable
	numrows	integer	counting variable
	pct2030	real	percentage
	pct3040	real	percentage
	pct4050	real	percentage
	pct5060	real	percentage
	pct6070	real	percentage
	pct7080	real	percentage
	pct8090	real	percentage
	pct9010	real	percentage
	pct90100	real	percentage
	pct<00	real	percentage
	pct<20	real	percentage
	pct<60	real	percentage
	pct<65	real	percentage
	pct<70	real	percentage
	pct<75	real	percentage
	pct<80	real	percentage
	pct<85	real	percentage
	pct<90	real	percentage

Used In	Name	Type	Description
intelrpt.prg	pct<95	real	percentage
	pct>00	real	percentage
	pct>100	real	percentage
	prdesc	text	= propdesc in BOATPROP
	proptst	text	= proptype in BOATPROP
	strtdate	date	starting date
	sumconst	integer	counting variable
	sumeng	integer	counting variable
	sumfuel	integer	counting variable
	sumhull	integer	counting variable
	sumprop	integer	counting variable
	sumtype	integer	counting variable
	sumuse	integer	counting variable
	totbrds	integer	counting variable
	totviols	integer	summation variable
	totviols	integer	counting variable
	tv1	integer	summation variable
	tv10	integer	summation variable
	tv2	integer	summation variable
	tv20	integer	summation variable
	tv3	integer	summation variable
	tv30	integer	summation variable
	tv4	integer	summation variable
	tv5	integer	summation variable
	typdesc1	text	= vtypedes in BOATTYPE
	typetest	text	= vtype in BOATTYPE
	usedesc1	text	= vusedesc in BOATUSE
	usetest	text	= vuse in BOATUSE
	vltndesc	text	= violdesc in VIOLS
	vno	text	= violno in VIOLS
	vpct2030	real	percentage
	vpct3040	real	percentage
	vpct4050	real	percentage
	vpct5060	real	percentage
	vpct6070	real	percentage
	vpct7080	real	percentage
	vpct8090	real	percentage
	vpct9010	real	percentage
	vpct<00	real	percentage
	vpct<20	real	percentage
	vpct<60	real	percentage
	vpct<65	real	percentage
	vpct<70	real	percentage
	vpct<75	real	percentage
	vpct<80	real	percentage
	vpct<85	real	percentage
	vpct<90	real	percentage
	vpct<95	real	percentage
	vpct>00	real	percentage
	vpct>100	real	percentage
	yr<1960	integer	counting variable
	yr<1965	integer	counting variable
	yr<1970	integer	counting variable
	yr<1975	integer	counting variable
	yr<1980	integer	counting variable
	yr<1985	integer	counting variable
	yr<1990	integer	counting variable
	yr<1995	integer	counting variable
	yr<2000	integer	counting variable
	yr>2000	integer	counting variable
modmain.prg	pick3	integer	menu variable

Used In	Name	Type	Description
mod_brd.prg	b1	date	bodate in boarding
	b14	text	viol54 in boarding
	b24	text	viol64 in boarding
	b45	text	bounit in boarding
	b10	text	obscity
	b11	text	obsstate in boarding
	b12	real	obslat in boarding
	b13	real	obslong in boarding
	b15	text	viol55
	b16	text	viol56 in boarding
	b17	text	viol57 in boarding
	b18	text	viol58 in boarding
	b19	text	viol59 in boarding
	b2	time	botime in boarding
	b20	text	viol61 in boarding
	b21	text	viol61 in boarding
	b22	text	viol62 in boarding
	b23	text	viol63 in boarding
	b25	text	viol65 in boarding
	b26	text	viol66 in boarding
	b27	text	viol67 in boarding
	b28	text	viol68 in boarding
	b29	text	uns69 in boarding
	b30	text	uns70 in boarding
	b31	text	uns71 in boarding
	b32	text	uns72 in boarding
	b33	text	uns73 in boarding
	b34	text	uns74 in boarding
	b35	text	uns75 in boarding
	b36	text	uns76 in boarding
	b37	text	uns77 in boarding
	b38	text	uns78 in boarding
	b39	text	uns79 in boarding
	b40	text	borem1 in boarding
	b41	text	borem2 in boarding
	b42	text	borem3 in boarding
	b43	text	boname in boarding
	b44	text	borate in boarding
	b46	text	bodist in boarding
	b47	text	boopfac in boarding
	b48	text	boday in boarding
	b49	time	bsunrise in boarding
	b5	integer	adultpfd in boarding
	b50	time	bsunset in boarding
	b51	text	bovhin in boarding
	b52	integer	weapons in boarding
	b6	integer	childpfd in boarding
	b7	text	owstatus in boarding
	b8	text	obsbody in boarding
	b9	text	obscount in boarding
	chk hin	text	flag variable
	chk name	text	flag variable
	correct	text	user input
	er	text	user input
	er	text	flag variable
	error	text	flag variable
	more	text	flag variable
	mrvsls	text	user input
	pg	integer	counting variable
	review	text	flag variable
	vadult	integer	adultpfd in boarding
	vbodate	date	boarding date
	vboday	text	boday in boarding
	vbodist	text	bodist in boarding
	vboname	text	boname in boarding

Used In	Name	Type	Description
mod_brd.prg	vbono	text	boarding report number
	vboopfac	text	boopfac in boarding
	vborate	text	borate in boarding
	vborem1	text	borem1 in boarding
	vborem2	text	borem2 in boarding
	vborem3	text	borem3 in boarding
	vbosrise	time	bsunrise in boarding
	vbosset	time	bsunset in boarding
	vbotime	time	boarding time
	vbounit	text	bounit in boarding
	vchild	integer	childpfd in baording
	vobcount	text	obscount in boarding
	vobsbody	text	obsbody in boarding
	vobscity	text	obscity in boarding
	vobslat	real	latitude position
	vobslong	real	longitude position
	vobstate	text	obsstate in boarding
	vostatus	text	owstatus in boarding
	vuns69	text	unsafe condition 69
	vuns70	text	unsafe condition 70
	vuns71	text	unsafe condition 71
	vuns72	text	unsafe condition 72
	vuns73	text	unsafe condition 73
	vuns74	text	unsafe condition 74
	vuns75	text	unsafe condition 75
	vuns76	text	unsafe condition 76
	vuns77	text	unsafe condition 77
	vuns78	text	unsafe condition 78
	vuns79	text	unsafe condition 79
	vviol54	text	violation code 54
	vviol55	text	violation code 55
	vviol56	text	violation code 56
	vviol57	text	violation code 57
	vviol58	text	violation code 58
	vviol59	text	violation code 59
	vviol60	text	violation code 60
	vviol61	text	violation code 61
	vviol62	text	violation code 62
	vviol63	text	violation code 63
	vviol64	text	violation code 64
	vviol65	text	violation code 65
	vviol66	text	violation code 66
	vviol67	text	violation code 67
	vviol68	text	violation code 68
	vvslhin	text	vslhin in vessel
	vvslpob	integer	personnel on board
	vweapons	integer	weapons in boarding
mod_owop.prg	chk ownr	text	flag variable
	choice	text	user input variable
	choice	integer	user input
	er	text	flag variable
	error	text	flag variable
	error	text	user input
	go	text	user input
	nmchange	text	flag variable
	op1	text	oplname in operator
	op10	text	opcourse in operator
	op11	integer	opid in operator
	op2	text	opfname in operator
	op3	text	opmi in operator
	op4	text	opaddr in operator
	op5	text	opcity in operator
	op6	text	opstate in operator
	op7	text	opzip in operator
	op8	text	optelno in operator

Used In	Name	Type	Description
mod_owop.prg	op9	date	opdob in operator
	opchange	text	flag variable
	opfound	text	flag variable
	opid	integer	operator id (system)
	ow1	text	ownlname in owners
	ow10	text	ownid in owners
	ow2	text	ownfname in owners
	ow3	text	ownmi in owners
	ow4	text	ownaddr in owners
	ow5	text	owncity in owners
	ow6	text	ownstate in owners
	ow7	text	ownzip in owners
	ow8	text	owntelno in owners
	ow9	date	owndob in owners
	owchange	text	flag variable
	owfound	text	flag variable
	vopaddr	text	operator street address
	vopcity	text	operator city address
	vopcrse	text	operator course code
	vopdob	date	operator date of birth
	vopfname	text	operator first name
	vopid	integer	operator id (system)
	voplname	text	operator last name
	vopmi	text	operator middle initial
	vopst	text	operator state address
	voptelno	text	operator telephone number
	vopzip	text	operator zip code
	vowaddr	text	owner street address
	vowcity	text	owner city address
	vowdob	date	owner date of birth
	vowfname	text	owner first name
	vowlname	text	owner last name
	vowmi	text	owner middle initial
	vownid	text	owner id (system)
	vownid	integer	owner id (system)
	vowst	text	owner state address
	vowtelno	text	owner telephone number
	vowzip	text	owner zip code
mod_sar.prg	change	text	flag variable
	correct	text	user input
	er	text	user input
	more	text	flag variable
	mrsar	text	user input
	oldfold	text	folder number
	reenter	text	flag variable
	s1	text	folderno in sar
	s10	integer	sarpob in sar
	s11	text	sarvhin in sar
	s12	real	sarlat in sar
	s13	real	sarlong in sar
	s14	integer	distoff in sar
	s15	text	immedsar in sar
	s16	text	winddir in sar
	s17	text	windspd in sar
	s18	text	visdist in sar
	s19	text	clouds in sar
	s2	text	ucn in sar
	s20	text	ceiling in sar
	s21	text	seas in sar
	s22	text	swells in sar
	s3	text	mucn in sar
	s4	date	sardate in sar
	s5	text	sarday in sar
	s6	time	sartime in sar
	s7	time	ssunrise in sar

Used In	Name	Type	Description
mod_sar.prg	s8	time	ssunset
	s9	text	sarnod in sar
	vceiling	text	ceiling in sar
	vdistoff	integer	distance offshore
	vfoldno	text	sar folder number
	vimmed	text	immediate sar?
	vmucn	text	multi-unit case number
	vsardate	date	sar date
	vsarday	text	sar day of the week
	vsarhin	text	sarvhin in sar
	vsarlat	real	latitude position
	vsarlong	real	longitude position
	vsarnod	text	nature of distress
	vsarpob	integer	personnel on board
	vsartime	time	sar time
	vseas	text	seas in sar
	vsunrise	time	bsunrise in boarding
	vsunset	time	bsunset in boarding
	vswells	text	swells in sar
	vucn	text	unit case number
	vvisdist	text	visdist in sar
	vwinddir	text	winddir in sar
	vwindspd	text	windspd in sar
mod_sut.prg	both	text	flag variable
	change	text	flag variable
	correct	text	user input
	done	text	flag variable
	foldok	text	flag variable
	more	text	flag variable
	mrsut	text	user input
	name	text	flag variable
	reenter	text	user input
	su10	date	alngdate in sarunits
	su11	time	towtime in sarunits
	su12	date	towdate in sarunits
	su13	time	dvmoored in sarunits
	su14	date	dvmdate in sarunits
	su15	time	sutmoor in sarunits
	su16	date	sutdate in sarunits
	su1	text	foldno in sarunits
	su2	text	unit in sarunits
	su3	time	uwtime in sarunits
	su4	date	uwdate
	su5	time	oseta in sarunits
	su6	date	etadate in sarunits
	su7	time	ostime in sarunits
	su8	date	osdate in sarunits
	su9	text	alngtime in sarunits
	valgdate	date	date alongside
	valgtime	time	time alongside
	vdateos	date	on scene date
	vdvmoord	date	date distressed vessel mo
	vdvmoort	time	time distressed vessel mo
	vetadate	date	on scene eta date
	vfoldno	text	sar folder number
	voseta	time	on scene eta time
	vostime	time	on scene time
	vsumoord	date	date sar unit moored
	vsumoort	time	time sar unit moored
	vtowdate	date	date started towing
	vtowtime	time	time started towing
	vunit	text	sar unit name
	vuwdate	date	underway date
	vuwtime	time	underway time

Used In	Name	Type	Description
mod_vhin.prg	chk hin	text	flag variable
	chk name	text	flag variable
	correct	text	user input
	correct	text	flag variable
	er	text	user input
	error	text	user input
	foundv	text	user input
	more	text	flag variable
	mrvsls	text	user input
	vbadhin	text	vslhin in vessel
	vconstr	text	coded vessel construction
	voffno	text	vessel official numbers
	vslfound	text	flag variable
	vvsleng	text	coded vessel engine compa
	vvslft	integer	vessel length (feet)
	vvslfuel	text	coded fuel compartment
	vvslhin	text	vslhin in vessel
	vvslhp	text	vessel horsepower
	vvslhull	text	coded vessel hull materia
	vvslin	integer	vessel length (inches)
	vvslmake	text	vessel make
	vvslmod	text	vslmodel in vessel
	vvslname	text	vessel name
	vvslno	text	vessel number
	vvsl rop	text	coded vessel propulsion
	vvslton	integer	vsltons in vessel
	vvsltype	text	coded vessel type
	vvsluse	text	coded vessel use
	vvslyear	text	vessel year
operator.prg	datediff	integer	date difference
	enddate	date	ending date
	endltr	text	ending letter
	ltrdiff	integer	letter difference
	numrows	integer	counting variable
	strtdate	date	starting date
	strtltr	text	starting letter
opervsl.prg	allrecs	text	flag - all records?
	endltr	text	ending letter
	ltrdiff	integer	letter difference
	strtltr	text	start letter
opervsls.prg	allrecs	text	user all records input
	endltr	text	ending letter
	ltrdiff	integer	letter difference
	numrows	integer	counting variable
	strtltr	text	starting letter
owners.prg	datediff	integer	date difference
	enddate	date	ending date
	endltr	text	ending letter
	ltrdiff	integer	letter difference
	numrows	integer	counting variable
	strtdate	date	starting date
	strtltr	text	starting letter
ownvsl.prg	allrecs	text	flag --- all records?
	allrecs	text	user all records input
	endltr	text	ending letter
	ltrdiff	integer	letter difference
	numrows	integer	counting variable
	strtltr	text	starting letter
packdbms.prg	choice	text	user choice input
perintel.prg	datediff	integer	date difference
	enddate	date	ending date
	endltr	text	ending letter
	ltrdiff	integer	letter difference
	numrows	integer	counting variable
	strtdate	date	starting date
	strtltr	text	starting letter

Used In	Name	Type	Description
prs_main.prg	pick9	integer	menu variable
qrymain.prg	pick10	integer	menu variable
quer_per.prg	bohisprn	text	user input
	chk-op	text	flag variable
	count	integer	counting variable
	done	text	flag variable
	more	text	flag variable
	mrname	text	user input
	opid	integer	operator id (system)
	oprfound	text	flag variable
	owndone	text	flag variable
	ownfound	text	flag variable
	owoptest	text	flag variable
	perintpr	text	user input
	persprn	text	user input
	sarhsprn	text	user input
	searchok	text	flag variable
	status	text	flag variable
	valc	text	operator alcohol conditio
	vbodate	date	boarding date
	vbolat	real	latitude position
	vbolong	real	longitude position
	vboname	text	boname in boarding
	vbono	text	boarding report number
	vborate	text	borate in boarding
	vborem1	text	borem1 in boarding
	vborem2	text	borem2 in boarding
	vborem3	text	borem3 in boarding
	vbotime	time	boarding time
	vdate	date	intelligence date
	vdistoff	integer	distance offshore
	vepic	text	epic code
	vfname	text	first name
	vfoldno	text	sar folder number
	vimmed	text	immediate sar?
	vintdate	date	vessel intelligence date
	vlname	text	last name
	vmi	text	middle initial
	vmucn	text	multi-unit case number
	voffno	text	vessel official numbers
	vopaddr	text	operator street address
	vopcity	text	operator city address
	vopcrse	text	operator course code
	vopdob	date	operator date of birth
	vopfname	text	operator first name
	voplname	text	operator last name
	vopmi	text	operator middle initial
	vopst	text	operator state address
	voptelno	text	operator telephone number
	vopzip	text	operator zip code
	vowid	integer	owner record id number
	vperaddr	text	street address
	vpercity	text	city address
	vperdob	date	date of birth
	vpermi	text	middle initial
	vperst	text	state address
	vpertel	text	telephone number
	vperzip	text	zip code
	vpfname	text	first name
	vplname	text	last name
	vrem1	text	personnel intelligence re
	vrem2	text	personnel intelligence re
	vrem3	text	personnel intelligence re
	vsardate	date	sar date
	vsarday	text	sar day of the week

Used In	Name	Type	Description
quer_per.prg	vsarlat	real	latitude position
	vsarlong	real	longitude position
	vsarpob	integer	personnel on board
	vsartime	time	sar time
	vslprn	text	user input
	vstolen	text	stolen vessel
	vucn	text	unit case number
	vuns69	text	unsafe condition 69
	vuns70	text	unsafe condition 70
	vuns71	text	unsafe condition 71
	vuns72	text	unsafe condition 72
	vuns73	text	unsafe condition 73
	vuns74	text	unsafe condition 74
	vuns75	text	unsafe condition 75
	vuns76	text	unsafe condition 76
	vuns77	text	unsafe condition 77
	vuns78	text	unsafe condition 78
	vuns79	text	unsafe condition 79
	vviol54	text	violation code 54
	vviol55	text	violation code 55
	vviol56	text	violation code 56
	vviol57	text	violation code 57
	vviol58	text	violation code 58
	vviol59	text	violation code 59
	vviol60	text	violation code 60
	vviol61	text	violation code 61
	vviol62	text	violation code 62
	vviol63	text	violation code 63
	vviol64	text	violation code 64
	vviol65	text	violation code 65
	vviol66	text	violation code 66
	vviol67	text	violation code 67
	vviol68	text	violation code 68
	vvrem1	text	vessel intelligence remar
	vvrem2	text	vessel intelligence remar
	vvrem3	text	vessel intelligence remar
	vvslhin	text	vslhin in vessel
	vvslmake	text	vessel make
	vvslname	text	vessel name
	vvslno	text	vessel number
	vvslpob	integer	personnel on board
	vweapons	integer	weapons in boarding
quer_vsl.prg	bohisprn	text	user input
	chk-hin	text	flag variable
	chk-intl	text	flag variable
	chk-name	text	flag variable
	chk-no	text	flag variable
	chkoffno	text	flag variable
	choice	integer	user input choice
	count	integer	counting variable
	menu	text	flag variable
	more	text	flag variable
	nameok	text	flag variable
	opid	integer	operator id (system)
	owhisprn	text	user input
	owid	integer	owner id (system)
	sarhsprn	text	user input
	valc	text	operator alcohol conditio
	vbodate	date	boarding date
	vbolat	real	latitude position
	vbolong	real	longitude position
	vboname	text	boname in boarding
	vbono	text	boarding report number
	vborate	text	borate in boarding
	vborem1	text	borem1 in boarding

Used In	Name	Type	Description
quer_vsl.prg	vborem2	text	borem2 in boarding
	vborem3	text	borem3 in boarding
	vbotime	time	boarding time
	vdate	date	intelligence date
	vdistoff	integer	distance offshore
	vdob	date	date of birth
	vepic	text	epic code
	vfname	text	first name
	vfoldno	text	sar folder number
	vimmed	text	immediate sar?
	vintdate	date	vessel intelligence date
	vintprn	text	user input
	vlname	text	last name
	vmi	text	middle initial
	vmucn	text	multi-unit case number
	voffno	text	vessel official numbers
	vopaddr	text	operator street address
	vopcity	text	operator city address
	vopcrse	text	operator course code
	vopdob	date	operator date of birth
	vopfname	text	operator first name
	voplname	text	operator last name
	vopmi	text	operator middle initial
	vopst	text	operator state address
	voptelno	text	operator telephone number
	vopzip	text	operator zip code
	vowaddr	text	owner street address
	vowcity	text	owner city address
	vowdob	date	owner date of birth
	vowfname	text	owner first name
	vowlname	text	owner last name
	vowmi	text	owner middle initial
	vowst	text	owner state address
	vowtelno	text	owner telephone number
	vowzip	text	owner zip code
	vrem1	text	personnel intelligence re
	vrem2	text	personnel intelligence re
	vrem3	text	personnel intelligence re
	vsardate	date	sar date
	vsarday	text	sar day of the week
	vsarlat	real	latitude position
	vsarlong	real	longitude position
	vsarpob	integer	personnel on board
	vsartime	time	sar time
	vslprn	text	flag variable
	vstolen	text	stolen vessel
	vucn	text	unit case number
	vuns69	text	unsafe condition 69
	vuns70	text	unsafe condition 70
	vuns71	text	unsafe condition 71
	vuns72	text	unsafe condition 72
	vuns73	text	unsafe condition 73
	vuns74	text	unsafe condition 74
	vuns75	text	unsafe condition 75
	vuns76	text	unsafe condition 76
	vuns77	text	unsafe condition 77
	vuns78	text	unsafe condition 78
	vuns79	text	unsafe condition 79
	vviol54	text	violation code 54
	vviol55	text	violation code 55
	vviol56	text	violation code 56
	vviol57	text	violation code 57
	vviol58	text	violation code 58
	vviol59	text	violation code 59
	vviol60	text	violation code 60

Used In	Name	Type	Description
quer_vsl.prg	vviol61	text	violation code 61
	vviol62	text	violation code 62
	vviol63	text	violation code 63
	vviol64	text	violation code 64
	vviol65	text	violation code 65
	vviol66	text	violation code 66
	vviol67	text	violation code 67
	vviol68	text	violation code 68
	vvrem1	text	vessel intelligence remar
	vvrem2	text	vessel intelligence remar
	vvrem3	text	vessel intelligence remar
	vvslhin	text	vslhin in vessel
	vvslmake	text	vessel make
	vvslname	text	vessel name
	vvslno	text	vessel number
	vvslpob	integer	personnel on board
	vweapons	integer	weapons in boarding
rptmain.prg	pick4	integer	menu variable
sar_main.prg	pick7	integer	menu variable
sardates.prg	jundate1	date	time computation variable
sarday.prg	aprdate1	date	starting date
	aprdate2	date	ending date
	aprsar	integer	summation variable
	augdate1	date	starting date
	augdate2	date	ending date
	augsar	integer	summation variable
	change	real	percentage
	count	integer	counting variable
	decdate1	date	starting date
	decdate2	date	ending date
	decsar	integer	summation variable
	enddate	date	ending date
	febdate1	date	starting date
	febdate2	date	ending date
	febsar	integer	summation variable
	found	text	flag variable
	friday	integer	counting variable
	initdate	date	starting date
	jandate1	date	starting date
	jandate2	date	ending date
	jansar	integer	summation variable
	juldate1	date	starting date
	juldate2	date	ending date
	julsar	integer	summation variable
	jundate1	date	starting date
	jundate2	date	ending date
	junsar	integer	summation variable
	mardate1	date	starting date
	mardate2	date	ending date
	marsar	integer	summation variable
	maydate1	date	starting date
	maydate2	date	ending date
	maysar	integer	summation variable
	monday	integer	counting variable
	new	integer	this year's sar cases
	novdate1	date	starting date
	novdate2	date	ending date
	novsar	integer	summation variable
	numrows	integer	counting variable
	numyears	integer	counting variable
	octdate1	date	starting date
	octdate2	date	ending date
	octsar	integer	summation variable
	old	integer	previous year's sar cases
	pctapr	real	percentage

Used In	Name	Type	Description
sarday.prg	pctaug	real	percentage
	pctchang	real	percentage
	pctdec	real	percentage
	pctfeb	real	percentage
	pctfri	real	percentage
	pctjan	real	percentage
	pctjul	real	percentage
	pctjun	real	percentage
	pctmar	real	percentage
	pctmay	real	percentage
	pctmon	real	percentage
	pctnov	real	percentage
	pctoct	real	percentage
	pctsat	real	percentage
	pctsep	real	percentage
	pctsun	real	percentage
	pctthu	real	percentage
	pcttue	real	percentage
	pctwed	real	percentage
	sar00	integer	counting variable
	sar01	integer	counting variable
	sar02	integer	counting variable
	sar03	integer	counting variable
	sar04	integer	counting variable
	sar05	integer	counting variable
	sar06	integer	counting variable
	sar07	integer	counting variable
	sar08	integer	counting variable
	sar09	integer	counting variable
	sar10	integer	counting variable
	sar11	integer	counting variable
	sar12	integer	counting variable
	sar81	integer	counting variable
	sar82	integer	counting variable
	sar83	integer	counting variable
	sar84	integer	counting variable
	sar85	integer	counting variable
	sar86	integer	counting variable
	sar87	integer	counting variable
	sar88	integer	counting variable
	sar89	integer	counting variable
	sar90	integer	counting variable
	sar91	integer	counting variable
	sar92	integer	counting variable
	sar93	integer	counting variable
	sar94	integer	counting variable
	sar95	integer	counting variable
	sar96	integer	counting variable
	sar97	integer	counting variable
	sar98	integer	counting variable
	sar99	integer	counting variable
	saturday	integer	counting variable
	sepdate1	date	starting date
	sepdate2	date	ending date
	sepsar	integer	summation variable
	sum1	integer	summation variable
	sum10	integer	summation variable
	sum11	integer	summation variable
	sum12	integer	summation variable
	sum2	integer	summation variable
	sum20	integer	summation variable
	sum3	integer	summaticn variable
	sum4	integer	summation variable
	sum5	integer	summation variable
	sum6	integer	summation variable

Used In	Name	Type	Description
sarday.prg	sunday	integer	counting variable
	thursday	integer	counting variable
	totsars	integer	summation variable
	tuesday	integer	counting variable
	wedsday	integer	counting variable
	year	integer	saryear
	yr1981	date	starting date
	yr1981a	date	ending date
	yr1982	date	starting date
	yr1982a	date	ending date
	yr1983	date	starting date
	yr1983a	date	ending date
	yr1984	date	starting date
	yr1984a	date	ending date
	yr1985	date	starting date
	yr1985a	date	ending date
	yr1986	date	starting date
	yr1986a	date	ending date
	yr1987	date	starting date
	yr1987a	date	ending date
	yr1988	date	starting date
	yr1988a	date	ending date
	yr1989	date	starting date
	yr1989a	date	ending date
	yr1990	date	starting date
	yr1990a	date	ending date
	yr1991	date	starting date
	yr1991a	date	ending date
	yr1992	date	starting date
	yr1992a	date	ending date
	yr1993	date	starting date
	yr1993a	date	ending date
	yr1994	date	starting date
	yr1994a	date	ending date
	yr1995	date	starting date
	yr1995a	date	ending date
	yr1996	date	starting date
	yr1996a	date	ending date
	yr1997	date	starting date
	yr1997a	date	ending date
	yr1998	date	starting date
	yr1998a	date	ending date
	yr1999	date	starting date
	yr1999a	date	ending date
	yr2000	date	starting date
	yr2000a	date	ending date
	yr2001	date	starting date
	yr2001a	date	ending date
	yr2002	date	starting date
	yr2002a	date	ending date
	yr2003	date	starting date
	yr2003a	date	ending date
	yr2004	date	starting date
	yr2004a	date	ending date
	yr2005	date	starting date
	yr2005a	date	ending date
	yr2006	date	starting date
	yr2006a	date	ending date
	yr2007	date	starting date
	yr2007a	date	ending date
	yr2008	date	starting date
	yr2008a	date	ending date
	yr2009	date	starting date
	yr2009a	date	ending date
	yr2010	date	starting date

Used In	Name	Type	Description
sarday.prg	yr2010a	date	ending date
	yr2011	date	starting date
	yr2011a	date	ending date
	yr2012	date	starting date
	yr2012a	date	ending date
	yrtest1	integer	leap year test
	yrtest2	integer	leap year test
sardump.prg	datediff	integer	date difference
	enddate	date	ending date
	numrows	integer	counting variable
	strtdate	date	starting date
sarmajor.prg	datediff	integer	date difference
	easttst	real	nautical position coord.
	enddate	date	ending date
	gridcnt	integer	counting variable
	inarea	integer	counting variable
	inpct	real	percentage
	northtst	real	nautical position coord.
	numrows	integer	counting variable
	outarea	integer	counting variable
	outpct	real	percentage
	southtst	real	nautical position coord.
	strtdate	date	starting date
	temppct	real	percentage
	totsars	integer	counting variable
	westtst	real	nautical position coord.
sarminor.prg	datediff	integer	date difference
	easttst	real	nautical position coord.
	enddate	date	ending date
	gridcnt	integer	counting variable
	inarea	integer	counting variable
	inpct	real	percentage
	northtst	real	nautical position coord.
	numrows	integer	counting variable
	outarea	integer	counting variable
	outpct	real	percentage
	southtst	real	nautical position coord.
	strtdate	date	starting date
	temppct	real	percentage
	westtst	real	nautical position coord.
sarsmbt.prg	datediff	integer	date difference
	enddate	date	starting date
	inpct	real	percentage
	inpcttmp	real	percentage
	inside	integer	summation variable
	numrows	integer	counting variable
	outpct	real	percentage
	outpcttm	real	percentage
	outside	integer	summation variable
	strtdate	date	starting date
	totsars	integer	counting variable
sarsys.prg	pick1	integer	menu variable
sartimes.prg	algavg	real	average alongside time
	algtimes	integer	counting variable
	datediff	integer	date difference
	datedour	real	time computation variable
	enddate	date	ending date
	maxaltim	real	maximum alongside time
	maxostim	real	maximum on scene time
	maxshtim	real	maximum search time
	maxtttim	real	maximum total case time
	maxtwtim	real	maximum towing time
	minaltim	real	minimum alongside time
	minostim	real	minimum on scene time
	minshtim	real	minimum search time

Used In	Name	Type	Description
sartimes.prg	mintttim	real	minimum cotal case time
	mintwtim	real	minimum towing time
	oscavg	real	average on scene time
	osdate	date	on scene date
	ossum	real	summation variable
	ostime	time	on scene time
	ostimes	integer	counting variable
	schavg	real	average search time
	schtimes	integer	counting variable
	strtdate	date	starting date
	sutdate	date	sar unit moored date
	sutmoor	time	sar unit moored time
	test	real	comparison variable
	timeadd	real	time computation variable
	timehour	real	time computation variable
	totavg	real	average total case time
	totsum	real	summation variable
	tottimes	integer	counting variable
	towavg	real	average towing time
	towsum	real	summation variable
	towtimes	integer	counting variable
	uwdate	date	underway date
	uwtime	time	underway time
smboatrp.prg	datediff	integer	date difference
	enddate	date	ending date
	inside	integer	counting variable
	obslat	real	nautical position coord.
	obslong	real	nautical position coord.
	outside	integer	counting variable
	strtdate	date	starting date
	totbrds	integer	counting variable
startup.prg	chances	integer	chances to logon
	count	integer	counting variable
	username	text	user name input
	userpass	text	user password
	validusr	text	flag variable
utlmain.prg	pick11	integer	menu variable
vessel.prg	anytype	text	flag variable
	choice	text	user choice input
	endltr	text	ending letter
	legalch	text	flag variable
	legaltyp	text	flag variable
	legaluse	text	flag variable
	ltrdiff	integer	letter difference
	numrows	integer	counting variable
	strtltr	text	starting letter
	typchoic	text	user type choice
	typetest	text	vessel type
	usechoic	text	user vessel use choice
	usetest	text	vessel use
violfreq.prg	datediff	integer	date difference
	enddate	date	ending date
	pct54	real	percentage
	pct54tmp	real	percentage
	pct55	real	percentage
	pct55tmp	real	percentage
	pct56	real	percentage
	pct56tmp	real	percentage
	pct57	real	percentage
	pct57tmp	real	percentage
	pct58	real	percentage
	pct58tmp	real	percentage
	pct59	real	percentage
	pct59tmp	real	percentage
	pct60	real	percentage

Used In	Name	Type	Description
violfreq.prg	pct60tmp	real	percentage
	pct61	real	percentage
	pct61tmp	real	percentage
	pct62	real	percentage
	pct62tmp	real	percentage
	pct63	real	percentage
	pct63tmp	real	percentage
	pct64	real	percentage
	pct64tmp	real	percentage
	pct65	real	percentage
	pct65tmp	real	percentage
	pct66	real	percentage
	pct66tmp	real	percentage
	pct67	real	percentage
	pct67tmp	real	percentage
	pct68	integer	percentage
	pct68tmp	real	percentage
	pct69	real	percentage
	pct69tmp	real	percentage
	pct70	real	percentage
	pct70tmp	real	percentage
	pct71	real	percentage
	pct71tmp	real	percentage
	pct72	real	percentage
	pct72tmp	real	percentage
	pct73	real	percentage
	pct73tmp	real	percentage
	pct74	real	percentage
	pct74tmp	real	percentage
	pct75	real	percentage
	pct75tmp	real	percentage
	pct76	real	percentage
	pct76tmp	real	percentage
	pct77	real	percentage
	pct77tmp	real	percentage
	pct78	real	percentage
	pct78tmp	real	percentage
	pct79	real	percentage
	pct79tmp	real	percentage
	strtdate	date	starting date
	uns69s	integer	counting variable
	uns70s	integer	counting variable
	uns71s	integer	counting variable
	uns72s	integer	counting variable
	uns73s	integer	counting variable
	uns74s	integer	counting variable
	uns75s	integer	counting variable
	uns76s	integer	counting variable
	uns77s	integer	counting variable
	uns78s	integer	counting variable
	uns79s	integer	counting variable
	unstot1	integer	summation variable
	unstot2	integer	summation variable
	unstot3	integer	summation variable
	unstot4	integer	summation variable
	unstot5	integer	summation variable
	unstot6	integer	summation variable
	unstot7	integer	summation variable
	unstotal	integer	total unsafe conditions
	viol54s	integer	counting variable
	viol55s	integer	counting variable
	viol56s	integer	counting variable
	viol57s	integer	counting variable
	viol58s	integer	counting variable
	viol59s	integer	counting variable

Used In	Name	Type	Description
violfreq.prg	viol60s	integer	counting variable
	viol61s	integer	counting variable
	viol62s	integer	counting variable
	viol63s	integer	counting variable
	viol64s	integer	counting variable
	viol65s	integer	counting variable
	viol66s	integer	counting variable
	viol67s	integer	counting variable
	viol68s	integer	counting variable
	vtot1	integer	summation variable
	vtot10	integer	summation variable
	vtot11	integer	summation variable
	vtot12	integer	summation variable
	vtot13	integer	summation variable
	vtot2	integer	summation variable
	vtot3	integer	summation variable
	vtot4	integer	summation variable
	vtot5	integer	summation variable
	vtot6	integer	summation variable
	vtot7	integer	summation variable
	vtot8	integer	summation variable
	vtot9	integer	summation variable
	vtotal	integer	total violations
vsl_main.prg	pick8	integer	menu variable
vsloper.prg	allrecs	text	user all records input
	allrecs	text	flag - all records?
	endltr	text	ending letter
	ltrdiff	integer	letter difference
	numrows	integer	counting variable
	strtltr	text	starting letter
vslowner.prg	allrecs	text	flag - all records?
	allrecs	text	user all records input
	endltr	text	ending letter
	ltrdiff	integer	letter difference
	numrows	integer	counting variable
	strtltr	text	starting letter

Data Entry / Edit Rules

```
RULE  1   vslyear IN vessel fail
        or  vslyear IN vessel ge 0
    Message:*** Invalid Vessel Year ***

RULE  2   vslhp IN vessel fail
        or  vslhp IN vessel ge 0
    Message:*** Invalid Vessel Horsepower ***

RULE  3   vsluse IN vessel fail
        or  vsluse IN vessel eqa vuse IN boatuse
    Message:*** Invalid Vessel Use ***

RULE  4   vsltype IN vessel fail
        or  vsltype IN vessel eqa vtype IN boattype
    Message:*** Invalid Vessel Type ***

RULE  5   vslprop IN vessel fail
        or  vslprop IN vessel eqa proptype IN boatprop
    Message:*** Invalid Vessel Propulsion ***

RULE  6   vslhull IN vessel fail
        or  vslhull IN vessel eqa hmattype IN hullmat
    Message:*** Invalid Vessel hull Type ***

RULE  7   vsleng IN vessel fail
        or  vsleng IN vessel eqa ecomptyp IN engcomp
    Message:*** Invalid Engine Compartment Type ***

RULE  8   vslfuel IN vessel fail
        or  vslfuel IN vessel eqa fcomptyp IN fuelcomp
    Message:*** Invalid Fuel Compartment Type ***
```

```
RULE  9   vslconst IN vessel fail
       or  vslconst IN vessel eqa constype IN constr
   Message:*** Invalid Vessel Construction ***

RULE  10   cmeyear IN vessel fail
       or  cmeyear IN vessel ge 0
   Message:*** Invalid CME Decal Year ***

RULE  11   vslmasts IN vessel fail
       or  vslmasts IN vessel ge 0
   Message:*** Invalid No. Vessel Masts ***

RULE  12   sailno IN vessel fail
       or  sailno IN vessel ge 0
   Message:*** Invalid Sail Number ***

RULE  13   hlcolor IN vessel eqa shade IN colors
       and trcolor IN vessel eqa shade IN colors
       and slcolor IN vessel eqa shade IN colors
       and cbcolor IN vessel eqa shade IN colors
       and dkcolor IN vessel eqa shade IN colors
   Message:*** Invalid Color Entry ***

RULE  14   vsltons IN vessel > = 0
   Message:*** Invalid Vessel Tons ***

RULE  15   vsllenft IN vessel > = 0
   Message:*** Invalid Vessel Length (ft) ***

RULE  16   vsllenin IN vessel > = 0
       and vsllenin IN vessel < = 12
   Message:*** Invalid Vessel Length (in) ***

RULE  17   vslvalue IN vessel > = 0
   Message:*** Invalid Vessel Value ***
```

```
RULE 18    cmedecal IN vessel fail
        or  cmedecal IN vessel = Y
        or  cmedecal IN vessel = N
    Message:*** Invalid CME Decal Entry [Y/N] ***

RULE 19    opstate IN operator fail
        or  opstate IN operator eqa abbrev IN states
        or  opstate IN operator eqa symbol IN nations
    Message:*** Invalid Operator State/Nation ***

RULE 20    opzip IN operator fail
        or  opzip IN operator > = 0
    Message:*** Invalid Operator Zip Code ***

RULE 21    optelno IN operator fail
        or  optelno IN operator > = 0
    Message:*** Invalid Operator Phone Number ***

RULE 22    opcourse IN operator fail
        or  opcourse IN operator eqa coursetp IN courses
    Message:*** Invalid Operator Course Code ***

RULE 23    ownzip IN owners fail
        or  ownzip IN owners > = 0
    Message:*** Invalid Owner Zip Code ***

RULE 24    ownstate IN owners fail
        or  ownstate IN owners eqa symbol IN nations
        or  ownstate IN owners eqa abbrev IN states
    Message:*** Invalid Owner State/Nation ***

RULE 25    owntelno IN owners fail
        or  owntelno IN owners > = 0
    Message:*** Invalid Owner Phone Number ***
```

```
RULE  26    sarday IN sar fail
        or  sarday IN sar eqa dayabbr IN days
    Message:*** Invalid SAR Day of Week ***

RULE  27    sarpob IN sar > = 0
    Message:*** Invalid SAR POB ***

RULE  28    sarlat IN sar > = 0.0
        and sarlat IN sar < = 90.0
    Message:*** Invalid SAR Latitude ***

RULE  29    sarlong IN sar > = 0.0
        and sarlong IN sar < = 180.0
    Message:*** Invalid SAR Longitude ***

RULE  30    offshore IN sar > = 0
    Message:*** Invalid Distance Offshore ***

RULE  31    immedsar IN sar = Y
        or  immedsar IN sar = N
        or  immedsar IN sar fail
    Message:*** Invalid Immediate SAR Entry [Y/N] **

RULE  32    winddir IN sar > = 000
        and winddir IN sar < 360
    Message:*** Invalid Wind Direction ***

RULE  33    windspd IN sar > = 0
    Message:*** Invalid Wind Speed ***

RULE  34    visdist IN sar > = 0
    Message:*** Invalid Visual Distance ***
```

```
RULE 35   clouds IN sar > = 0
        and clouds IN sar < = 10
     Message:*** Invalid Cloud Percentage (Tenths) **

RULE 36   ceiling IN sar > = 0
     Message:*** Invalid Cloud Ceiling (Feet) ***

RULE 37   seas IN sar > = 0
     Message:*** Invalid Sea Height (Feet) ***

RULE 38   swells IN sar > = 0
     Message:*** Invalid Swell Height (Feet) ***

RULE 39   epiccode IN vslintel eqa epicno IN epic
        or  epiccode IN vslintel fail
     Message:*** Invalid EPIC Code Entry ***

RULE 40   folderno IN sar nea folderno IN sar
     Message:*** Duplicate SAR Folder Number ***

RULE 41   bono IN boarding nea bono IN boarding
     Message:*** Duplicate Boarding Report Number ***

RULE 42   vslpob IN boarding > = 0
     Message:*** Invalid Boarding POB ***

RULE 43   adultpfd IN boarding > = 0
     Message:*** Invalid Adult PFD/POB ***

RULE 44   childpfd IN boarding > = 0
     Message:*** Invalid Child PFD/POB ***

RULE 45   owstatus IN boarding eqa ostattyp IN status
        or  owstatus IN boarding fail
     Message:*** Invalid Owner Status ***
```

```
RULE 46   obsstate IN boarding eqa abbrev IN states
      or  obsstate IN boarding eqa symbol IN nations
      or  obsstate IN boarding fail
   Message:*** Invalid Boarding State/Nation ***

RULE 47   obslat IN boarding > = 0.0
      and obslat IN boarding < = 90.0
   Message:*** Invalid Boarding Latitude ***

RULE 48   obslong IN boarding > = 0.0
      and obslong IN boarding < = 180.0
   Message:*** Invalid Boarding Longitude ***

RULE 49   boday IN boarding eqa dayabbr IN days
      or  boday IN boarding fail
   Message:*** Invalid Boarding Day of Week ***

RULE 50   weapons IN boarding > = 0
   Message:*** Invalid Number of Weapons ***

RULE 51   bodist IN boarding eqa distno IN district
      or  bodist IN boarding fail
   Message:*** Invalid Boarding District ***

RULE 52   vslhin IN vessel nea vslhin IN vessel
   Message:*** Error: Duplicate Vessel HIN ***
```

APPENDIX D
MODULE FUNCTIONAL DESCRIPTIONS

The US Coast Guard Group Monterey SARELTIS prototype design includes eleven sets of application programs that access one central relational database. These sets of application programs include:

- SARSYS.PRG -- Main Menu Programs
- SENTRY.PRG -- Data Entry Menu and Programs
- MODIFY.PRG -- Data Modification Menu and Programs
- SREPORTS.PRG -- Reports Master Menu Programs
- INTRPTS.PRG -- Intelligence Reports Menu and Programs
- BRDRPTS.PRG -- Boarding Reports Menu and Programs
- SARRPTS.PRG -- Sar Reports Menu and Programs
- VSLRPTS.PRG -- Vessel Reports Menu and Programs
- PRSRPTS.PRG -- Personnel Reports Menu and Programs
- SQUERY.PRG -- Database Query Menu and Programs
- SUTILITY.PRG -- Utilities Menu and Programs

Each application program set consists of a master menu block and numerous modules that perform specific functions. The modules and menus are called Command and Menu files within the R:base 5000 programming language. Functional descriptions of each module, grouped by application program set, are contained in this appendix.

1. SARSYS.PRG

```
*(**********************************************************************
                       US COAST GUARD GROUP MONTEREY
            SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                       INTELLIGENCE SYSTEM (SARELTIS)
                       MAIN MENU and STARTUP APPLICATION

                    Author:    Jon D. Allen, LT, USCG
                                 Heidi R. Lang, LT, USN
                                 AUGUST 1987

*(**********************************************************************
*(***********************************************************************
     Program:          Access Input Module
     Author:           J D Allen and H R Lang
     Date Written:     August 1987
     Description:      This program calls the input master menu.
     Tables Used:      none
     Temp Tables:      none
     Reports Used:     none
     Forms Used:       none

*(***********************************************************************
     Program:          Access Modify Module
     Author:           J D Allen and H R Lang
     Date Written:     August 1987
     Description:      This program calls the modify master menu.
     Tables Used:      none
     Temp Tables:      none
     Reports Used:     none
     Forms Used:       none

*(***********************************************************************
     Program:          Access Query Module
     Author:           J D Allen and H R Lang
     Date Written:     August 1987
     Description:      This program calls the query master menu.
     Tables Used:      none
     Temp Tables:      none
     Reports Used:     none
     Forms Used:       none

*(***********************************************************************
     Program:          Access Reports Module
     Author:           J D Allen and H R Lang
     Date Written:     August 1987
     Description:      This program calls the reports master menu.
     Tables Used:      none
     Temp Tables:      none
     Reports Used:     none
     Forms Used:       none

*(***********************************************************************
     Program:          Access Utilities Module
     Author:           J D Allen and H R Lang
     Date Written:     August 1987
     Description:      This program calls the utility master menu.
     Tables Used:      none
     Temp Tables:      none
     Reports Used:     none
     Forms Used:       none

*(***********************************************************************
     Program:          Access Ad Hoc Module
     Author:           J D Allen and H R Lang
     Date Written:     August 1987
     Description:      This program allows the user to exit the
                       menus to the R:base 5000 command prompt to
                       execute ad hoc queries.
     Tables Used:      none
     Temp Tables:      none
     Reports Used:     none
     Forms Used:       none

***********************************************************************)
```

2. SENTRY.PRG

```
*(**********************************************************************
                         US COAST GUARD GROUP MONTEREY
          SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                         INTELLIGENCE SYSTEM (SARELTIS)
                      INPUT MASTER MENU and INPUT PROGRAMS

                    Author:    Jon D. Allen, LT, USCG
                                 Heidi R. Lang, LT, USN
                                 AUGUST 1987

*(**********************************************************************
     Program:           Add Sar
     Author:            H R Lang and J D Allen
     Date Written:      July 1987
     Description:       This program allows the user to input
                        sar case information.
     Tables Used:       vessel, sar, owners, operators, owvsl, opvsl
     Temp Tables:       none
     Reports Used:      none
     Forms Used:        sarform1, sarform2, sarform3, sarform4,
                        sarform5, sarform6, sarform7, sarform8,
                        sarform9

*(**********************************************************************
     Program:           Add Sar with Boarding
     Author:            H R Lang and J D Allen
     Date Written:      July 1987
     Description:       This program allows the user to enter sar
                        and boarding information.
     Tables Used:       boarding, vessel, owners, operator, owvsl,
                        opvsl, sar
     Temp Tables:       none
     Reports Used:      none
     Forms Used:        cg4100a1, cg4100a2, cg4100a3, cg4100a4,
                        cg4100a5, cg4100a6, cg4100a7, cg4100a8,
                        cg4100a9, sbdform1, sarform1, sarform8,
                        sarform9, sarform5

*(**********************************************************************
     Program:           Add Sar Units
     Author:            H R Lang and J D Allen
     Date Written:      August 1987
     Description:       This program allows the user to input sar
                        unit data.
     Tables Used:       sar, sarunits
     Temp Tables:       none
     Reports Used:      none
     Forms Used:        sarform8

*(**********************************************************************
*(********************************************************************
     Program:           Add Boarding
     Author:            H R Lang and J D Allen
     Date Written:      July 1987
     Description:       This program allows the user to input
                        boarding information.
     Tables Used:       boarding, vessel, owners, operator, owvsl,
                        opvsl,
     Temp Tables:       none
     Reports Used:      none
     Forms Used:        cg4100a1, cg4100a2, cg4100a3, cg4100a4,
                        cg4100a5, cg4100a6, cg4100a7, cg4100a8,
                        cg4100a9, sarform5

*(********************************************************************
     Program:           Add Owner
     Author:            H R Lang and J D Allen
     Date Written:      July 1987
     Description:       This program allows the user to input
                        additional owner information.
     Tables Used:       vessel, owners, owvsl
     Temp Tables:       none
     Reports Used:      none
     Forms Used:        ownrmod1, ownrmod2

*********************************************************************)
```

3. MODIFY.PRG

```
*(********************************************************************
                    US COAST GUARD GROUP MONTEREY
          SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                    INTELLIGENCE SYSTEM (SARELTIS)
                   MODIFY MAIN MENU and APPLICATIONS

                 Author:    Jon D. Allen, LT, USCG
                              Heidi R. Lang, LT, USN
                              AUGUST 1987

*(********************************************************************
     Program:           Modify Sar
     Author:            H R Lang and J D Allen
     Date Written:      August 1987
     Description:       This program allows the user to modify
                        sar information in the database.
     Tables Used:       sar, vessel
     Temp Tables:       none
     Reports Used:      none
     Forms Used:        sarmod

*(********************************************************************
*(*********************************************************************
     Program:          Modify Boarding
     Author:           H R Lang and J D Allen
     Date Written:     August 1987
     Description:      This program allows the user to modify
                       boarding information in the database.
     Tables Used:      boarding, vessel
     Temp Tables:      none
     Reports Used:     none
     Forms Used:       bomod1, bomod2, bomod3

*(*********************************************************************
     Program:          Modify Vessel
     Author:           H R Lang and J D Allen
     Date Written:     August 1987
     Description:      This program allows the user to modify
                       vessel information in the database.
     Tables Used:      vessel
     Temp Tables:      none
     Reports Used:     none
     Forms Used:       vslmod1, vslmod2

*(*********************************************************************
     Program:          Modify Owner and Operator
     Author:           H R Lang and J D Allen
     Date Written:     August 1987
     Description:      This program allows the user to modify
                       owner and operator information in the
                       database.
     Tables Used:      owners, operator, opown
     Temp Tables:      none
     Reports Used:     none
     Forms Used:       ownrmod1, ownrmod2, opermod1, opermod2

*(*********************************************************************
     Program:          Modify Sar Unit
     Author:           H R Lang and J D Allen
     Date Written:     August 1987
     Description:      This program allows the user to modify
                       sar unit data in the database.
     Tables Used:      sar, sarunits
     Temp Tables:      none
     Reports Used:     none
     Forms Used:       sunitmod

*(*********************************************************************
     Program:          Modify Vessel HIN
     Author:           H R Lang and J D Allen
     Date Written:     August 1987
     Description:      This program allows the user to modify
                       vessel hull identification numbers in
                       the database.
     Tables Used:      vessel, sar, boarding, owvsl, opvsl
     Temp Tables:      none
     Reports Used:     none
     Forms Used:       vhinform

*(*********************************************************************
*(*********************************************************************
     Program:         Add and Modify Vessel Intelligence
     Author:          H R Lang and J D Allen
     Date Written:    August 1987
     Description:     This program allows the user to input
                      and modify vessel intelligence
                      information.
     Tables Used:     vslintel
     Temp Tables:     none
     Reports Used:    none
     Forms Used:      vintform

*(*********************************************************************
     Program:         Add and Modify Personnel Intelligence
     Author:          H R Lang and J D Allen
     Date Written:    August 1987
     Description:     This program allows the user to input
                      and modify personnel intelligence
                      information.
     Tables Used:     perintel
     Temp Tables:     none
     Reports Used:    none
     Forms Used:      pintform

*********************************************************************)
```

4. SREPORTS.PRG

```
*(*********************************************************************

                      US COAST GUARD GROUP MONTEREY
          SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                      INTELLIGENCE SYSTEM (SARELTIS)
                   REPORTS MAIN MENU APPLICATION PROGRAM

                   Author:    Jon D. Allen, LT, USCG
                                Heidi R. Lang, LT, USN
                                AUGUST 1987

*(*********************************************************************
     Program:         Access Intelligence Reports
     Author:          J D Allen and H R Lang
     Date Written:    August 1987
     Description:     This program calls the intelligence reports
                      master menu and application programs.
     Tables Used:     none
     Temp Tables:     none
     Reports Used:    none
     Forms Used:      none

*(*********************************************************************
     Program:         Access Boarding Reports
     Author:          J D Allen and H R Lang
     Date Written:    August 1987
     Description:     This program calls the boarding reports
                      master menu and application programs.
     Tables Used:     none
     Temp Tables:     none
     Reports Used:    none
     Forms Used:      none

*(*********************************************************************
*(**********************************************************************
     Program:             Access Sar Reports
     Author:              J D Allen and H R Lang
     Date Written:        August 1987
     Description:         This program calls the SAR reports master
                          menu and application programs.
     Tables Used:         none
     Temp Tables:         none
     Reports Used:        none
     Forms Used:          none

*(**********************************************************************
     Program:             Access Vessel Reports
     Author:              J D Allen and H R Lang
     Date Written:        August 1987
     Description:         This program calls the vessel reports
                          master menu and application programs.
     Tables Used:         none
     Temp Tables:         none
     Reports Used:        none
     Forms Used:          none

*(**********************************************************************
     Program:             Access Personnel Reports
     Author:              J D Allen and H R Lang
     Date Written:        August 1987
     Description:         This program calls the personnel reports
                          master menu and application programs.
     Tables Used:         none
     Temp Tables:         none
     Reports Used:        none
     Forms Used:          none

***********************************************************************)
```

5. INTRPTS.PRG

```
*(**********************************************************************
                          US COAST GUARD GROUP MONTEREY
           SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                          INTELLIGENCE SYSTEM (SARELTIS)
            INTELLIGENCE REPORTS MENU and APPLICATION PROGRAMS

                    Author:    Jon D. Allen, LT, USCG
                                 Heidi R. Lang, LT, USN
                                 AUGUST 1987

*(**********************************************************************
     Program:             Personnel Intelligence Dump Program
     Author:              J D Allen and H R Lang
     Date Written:        August 1987
     Description:         This program performs a listing by date and
                          by alphabtic search of those personnel in
                          the table PERINTEL (personnel intelligence).
     Tables Used:         perintel
     Temp Tables:         persum
     Reports Used:        perintel
     Forms Used:          none

*(**********************************************************************
*(*****************************************************************
     Program:          EPIC Intelligence Analysis
     Author:           J D Allen and H R Lang
     Date Written:     August 1987
     Description:      Performs a dump of all vessels in the
                       VSLINTEL table within a specified time
                       period.  Four reports are generated
                       providing listings by EPIC Code, by
                       stolen vessels, and by intelligence
                       remarks, and a frequency breakdown
                       and analysis of EPIC Codes.
     Tables Used:      vslintel
     Temp Tables:      epicsum, stolesum, intelsum, epictmp1,
                       epictmp2, epictmp
     Reports Used:     epicsum, stolesum, intelsum, epicrpt
     Forms Used:       none

*(*****************************************************************
     Program:          Violation Vessel Profile Intelligence
     Author:           J D Allen and H R Lang
     Date Written:     August 1987
     Description:      This program examines vessel boardings to
                       create a profile of vessel which violate
                       specific laws during a specified time period.
     Tables Used:      boarding, vessel, viols, boattype, boatuse,
                       constr, engcomp, fuelcomp, hullmat, boatprop
     Temp Tables:      bosum, vslsum, violsum, vslmo,
                       summary1, summary2, summary3, summary4,
                       summary5, summary6, summary7, temp
     Reports Used:     vtypemo, vusemo, vconstmo, vengmo,
                       vfuelmo, vhullmo, vpropmo, vlenftmo,
                       vyearmo
     Forms Used:       none

******************************************************************)
```

6. BRDRPTS.PRG

```
*(*****************************************************************
                    US COAST GUARD GROUP MONTEREY
          SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                    INTELLIGENCE SYSTEM (SARELTIS)
             BOARDING REPORTS MENU and APPLICATION PROGRAMS

                  Author:    Jon D. Allen, LT, USCG
                               Heidi R. Lang, LT, USN
                               AUGUST 1987

*(*****************************************************************
     Program:          Violation / Unsafe Condition Frequency Analysis
     Author:           J D Allen and H R Lang
     Date Written:     August 1987
     Description:      This program performs a statistical analysis
                       of boarding violations and unsafe conditions
                       within a specified time period.
     Tables Used:      boarding
     Temp Tables:      frqchart, temp
     Reports Used:     violfreq, unsfreq
     Forms Used:       none

*(*****************************************************************
*(*********************************************************************
     Program:            Boarding Summary Listing Program
     Author:             J D Allen and H R Lang
     Date Written:       August 1987
     Description:        This program performs a listing of the
                         BOARDING table during a specified time
                         period.  The report is sorted by date
                         before being printed.
     Tables Used:        boarding
     Temp Tables:        bosum
     Reports Used:       boarding
     Forms Used:         none

*(*********************************************************************
     Program:            Boarding Major Grid Area Frequency Program
     Author:             J D Allen and H R Lang
     Date Written:       August 1987
     Description:        This program performs a statistical analysis
                         of Boardings to determine percentage of cases
                         which occur in areas specified by coordinates
                         found in the MAJAREA table during a specified
                         time period.
     Tables Used:        majarea, boarding
     Temp Tables:        bosum, positfrq
     Reports Used:       bomajor
     Forms Used:         none

*(*********************************************************************
     Program:            Boarding Minor Grid Area Frequency Program
     Author:             J D Allen and H R Lang
     Date Written:       August 1987
     Description:        This program performs a statistical analysis
                         of Boardings to determine percentage of
                         cases which occur in a specified location
                         defined in the MINAREA table during a
                         specified time period.
     Tables Used:        boarding, minarea
     Temp Tables:        bosum, positfrq
     Reports Used:       bominor
     Forms Used:         none

*(*********************************************************************
     Program:            Boarding Small Boat Area Frequency Program
     Author:             J D Allen and H R Lang
     Date Written:       August 1987
     Description:        This program performs a statistical analysis
                         of Boardings to determine percentage of cases
                         which occur in the small boat operating as
                         defined by Coast Guard Group Monterey's
                         OPLAN during a specified time period.
     Tables Used:        boarding
     Temp Tables:        temp, bosum
     Reports Used:       smboatrp
     Forms Used:         none

*(*********************************************************************
*(*************************************************************************
     Program:          Courtesy Motorboat Examinations (CME's)
                       Statistical Analysis
     Author:           J D Allen and H R Lang
     Date Written:     August 1987
     Description:      This program performs a statistical analysis
                       of vessels that were boarded to determine
                       the frequency of CME's and their year of
                       issue during a specified time period.
     Tables Used:      boarding, vessel
     Temp Tables:      bosum, vslsum, bovslsum, temp
     Reports Used:     cmerpt
     Forms Used:       none

*(*************************************************************************
     Program:          Daytime vs. Nighttime Frequency Analysis
     Author:           J D Allen and H R Lang
     Date Written:     August 1987
     Description:      This program performs a statistical analysis
                       of SAR cases and Boardings to determine
                       percentage of cases which occur in the
                       daytime vs those that occur at night during
                       a specified time period.
     Tables Used:      sar, boarding
     Temp Tables:      sarsum, bosum, temp
     Reports Used:     daynight
     Forms Used:       none

*************************************************************************)
```

7. SARRPTS.PRG

```
*(*************************************************************************
                         US COAST GUARD GROUP MONTEREY
              SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                       INTELLIGENCE SYSTEM (SARELTIS)
                    SAR REPORTS MENU and APPLICATION PROGRAMS

                      Author:    Jon D. Allen, LT, USCG
                                   Heidi R. Lang, LT, USN
                                   AUGUST 1987

*(*************************************************************************
     Program:          SAR Major Grid Area Frequency Analysis
     Author:           J D Allen and H R Lang
     Date Written:     August 1987
     Description:      This program performs a statistical analysis
                       of SAR cases to determine percentage of
                       cases which occur in specified locations in
                       the table MAJAREA during a specified time
                       period.
     Tables Used:      majarea, sar
     Temp Tables:      sarsum, positfrq
     Reports Used:     sarmajor
     Forms Used:       none

*(*************************************************************************
*(*********************************************************************

     Program:           SAR Minor Grid Area Frequency Analysis
     Author:            J D Allen and H R Lang
     Date Written:      August 1987
     Description:       This program performs a statistical analysis
                        of SAR cases to determine percentage of
                        cases which occur in locations specified by
                        coordinates contained in the table MINAREA
                        during a specified time period.
     Tables Used:       sar, minarea
     Temp Tables:       sarsum, positfrq
     Reports Used:      sarminor
     Forms Used:        none

*(*********************************************************************

     Program:           SAR Case Small Boat Area Frequency Program
     Author:            J D Allen and H R Lang
     Date Written:      August 1987
     Description:       This program performs a statistical analysis
                        of SAR Cases to determine percentage of
                        cases which occur in the small boat operating
                        as defined by Coast Guard Group Monterey's
                        OPLAN during a specified time period.
     Tables Used:       sar
     Temp Tables:       temp, sarsum
     Reports Used:      sarsmbt
     Forms Used:        none

*(*********************************************************************

     Program:           SAR Case Units Time Expended Frequency
     Author:            J D Allen and H R Lang
     Date Written:      August 1987
     Description:       This program performs a statistical analysis
                        of SAR cases to determine total time
                        expended, time to being on scene, time to
                        being alongside, time involved in searches,
                        and time involved in towing.  These times are
                        analyzed to produce totals, averages,
                        minimums, and maximums for a specific time
                        period.
     Tables Used:       sarunits
     Temp Tables:       sartime1, sartime2, sartime3, sartime4,
                        sartime5, temp
     Reports Used:      sartimes
     Forms Used:        none

*(*********************************************************************

     Program:           SAR Days, SAR Months, SAR Years Frequency
                        Analysis
     Author:            J D Allen and H R Lang
     Date Written:      August 1987
     Description:       This program performs a statistical analysis
                        of SAR cases to determine percentage of
                        cases which occurred on specific days of the
                        week, months of the year and years during a
                        specified time period.
     Tables Used:       sar
     Temp Tables:       sarsum1, sarstat, sarsum, temp
     Reports Used:      sardates, sardays
     Forms Used:        none

*(*********************************************************************
*(**********************************************************************
     Program:            Daytime vs. Nighttime Frequency Analysis
     Author:             J D Allen and H R Lang
     Date Written:       August 1987
     Description:        This program performs a statistical analysis
                         of SAR cases and Boardings to determine
                         percentage of cases which occur in the
                         daytime vs those that occur at night during
                         a specified time period.
     Tables Used:        sar, boarding
     Temp Tables:        sarsum, bosum, temp
     Reports Used:       daynight
     Forms Used:         none

*(**********************************************************************
     Program:            SAR Summary Listing Program
     Author:             J D Allen and H R Lang
     Date Written:       August 1987
     Description:        This program performs a dump of the table SAR
                         of those SAR cases that occurred during a
                         specified time period.  The listing is sorted
                         by date prior to being printed.
     Tables Used:        sar
     Temp Tables:        sarsum
     Reports Used:       sar
     Forms Used:         none

*(**********************************************************************
     Program:            Immediate SAR Frequency Program
     Author:             J D Allen and H R Lang
     Date Written:       August 1987
     Description:        This program performs a statistical analysis
                         of SAR cases to determine percentage of cases
                         which are of immediate nature during a
                         specified time period.
     Tables Used:        sar
     Temp Tables:        sarsum, temp
     Reports Used:       immedsar
     Forms Used:         none

**********************************************************************)
```

8. VSLRPTS.PRG

```
*(**********************************************************************
                          US COAST GUARD GROUP MONTEREY
              SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                          INTELLIGENCE SYSTEM (SARELTIS)
                   VESSEL REPORTS MENU and APPLICATION PROGRAMS

                      Author:    Jon D. Allen, LT, USCG
                                   Heidi R. Lang, LT, USN
                                   AUGUST 1987

*(**********************************************************************
*(*************************************************************************

     Program:             Vessel Summary Listing Program
     Author:              J D Allen and H R Lang
     Date Written:        August 1987
     Description:         This program performs a dump of the VESSEL
                          table.  Listing entries are selected by
                          vessel type, vessel use, and by an
                          alphabetic search.
     Tables Used:         vessel, boattype, boatuse
     Temp Tables:         vslsum, vslsum1, vslsum2
     Reports Used:        vessel
     Forms Used:          none

*(*************************************************************************

     Program:             Vessel - Owner Dump Program
     Author:              J D Allen and H R Lang
     Date Written:        August 1987
     Description:         This program performs a listing sorted by
                          vessel name of those vessels which are
                          owned by a specified group of owners.  All
                          of the vessels in the VESSEL table may be
                          selected or only a group specified in an
                          alphabetic search.
     Tables Used:         owvsl, owners, vessel
     Temp Tables:         ownvsls, ownvsls1, ownvsls2
     Reports Used:        vslowner
     Forms Used:          none

*(*************************************************************************

     Program:             Vessel - Operator Dump Program
     Author:              J D Allen and H R Lang
     Date Written:        August 1987
     Description:         This program performs a listing sorted by
                          vessel name of those vessels which are
                          operated by a specified group of operators.
                          The entire list of vessels may be printed
                          or a specified group based on an alphabetic
                          search.
     Tables Used:         opvsl, operator, vessel
     Temp Tables:         opervsls, opvsls1, opvsls2
     Reports Used:        vsloper
     Forms Used:          none

*************************************************************************)
```

9. PRSRPTS.PRG

```
*(*************************************************************************

                      US COAST GUARD GROUP MONTEREY
           SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                      INTELLIGENCE SYSTEM (SARELTIS)

              PERSONNEL REPORTS MENU and APPLICATION PROGRAMS

                    Author:    Jon D. Allen, LT, USCG
                                 Heidi R. Lang, LT, USN
                                 AUGUST 1987

*(*************************************************************************
*(***********************************************************************

     Program:          Vessel Owners Table Dump
     Author:           J D Allen and H R Lang
     Date Written:     August 1987
     Description:      This program prints a sorted listing of
                       vessel owners contained in the database
                       based on the owner's last name and date
                       of birth.
     Tables Used:      owners
     Temp Tables:      ownsum1, ownsum2
     Reports Used:     owners
     Forms Used:       none

*(**********************************************************************

     Program:          Vessel Operator Listing
     Author:           J D Allen and H R Lang
     Date Written:     August 1987
     Description:      This program performs a listing of operators
                       of vessels contained in the table OPERATOR.
                       They are selected by date of birth and by
                       the first letter of the last name.  The
                       listing is sorted alphabetically before
                       being printed.
     Tables Used:      operator
     Temp Tables:      opsum1, opsum2
     Reports Used:     operator
     Forms Used:       none

*(***********************************************************************

     Program:          Operator - Vessel Program
     Author:           J D Allen and H R Lang
     Date Written:     August 1987
     Description:      This program performs a listing sorted by
                       operator last name of those vessels which
                       are used by specified operators.  Operators
                       may be selected by an alphabetic search or
                       may be selected in their entirety.
     Tables Used:      opvsl, operator, vessel
     Temp Tables:      opervsls, opvsls1, opvsls2
     Reports Used:     opervsls
     Forms Used:       none

*(***********************************************************************

     Program:          Owner - Vessel Dump Program
     Author:           J D Allen and H R Lang
     Date Written:     August 1987
     Description:      This program performs a listing sorted by
                       owner last name of those owners which own
                       specific vessels.  Owners may be selected
                       in their entirety or may be selected by
                       an alphabetic search.
     Tables Used:      owvsl, owners, vessel
     Temp Tables:      ownvsls, ownvsls1, ownvsls2
     Reports Used:     ownvsls
     Forms Used:       none

**********************************************************************)
```

10. SQUERY.PRG

```
*(*******************************************************************
                  US COAST GUARD GROUP MONTEREY
         SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                  INTELLIGENCE SYSTEM (SARELTIS)
                 QUERY MENU and APPLICATION PROGRAMS

              Author:    Jon D. Allen, LT, USCG
                           Heidi R. Lang, LT, USN
                           AUGUST 1987

*(*******************************************************************
    Program:          Query Vessel
    Author:           H R Lang and J D Allen
    Date Written:     August 1987
    Description:      This program searches database for all
                      information concerning a particular vessel.
    Tables Used:      vessel, sar, boarding, owner, operator,
                      owvsl, opvsl, vslintel, perintel
    Temp Tables:      temp, vslint, perint, bohis, sarhis
    Reports Used:     vslqprn, boqprn, sarqprn, owqprn, opqprn,
                      pintqprn, vintqprn
    Forms Used:       vslquery, brdquery, sarquery, ownquery,
                      opquery, pintquer, vintquer

*(*******************************************************************
    Program:          Query Personnel
    Author:           H R Lang and J D Allen
    Date Written:     August 1987
    Description:      This program searches database for all
                      information concerning a particular person,
                      including all vessels owned or operated
                      and all historical information concerning
                      these vessels.
    Tables Used:      vessel, sar, boarding, owner, operator,
                      owvsl, opvsl, opown, vslintel, perintel
    Temp Tables:      temp, vslint, perint, bohis, sarhis,
                      vslsown, vslsopr, wvsljoin, pvsljoin
    Reports Used:     vslqprn, boqprn, sarqprn, perqprn,
                      pintqprn, vintqprn
    Forms Used:       vslquery, brdquery, sarquery, perquer1,
                      perquer2, pintquer, vintquer

*******************************************************************)
```

11. SUTILITY.PRG

```
*(***********************************************************************
                          US COAST GUARD GROUP MONTEREY
           SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                          INTELLIGENCE SYSTEM (SARELTIS)
                    UTILITIES MENU and APPLICATION PROGRAMS

                     Author:    Jon D. Allen, LT, USCG
                                  Heidi R. Lang, LT, USN
                                  AUGUST 1987

*(***********************************************************************
     Program:              Initialize SARSYS
     Author:               J D Allen and H R Lang
     Date Written:         August 1987
     Description:          This program removes all temporary
                           tables that can be created by the system.
     Tables Used:          none
     Temp Tables:          all
     Reports Used:         none
     Forms Used:           none

*(***********************************************************************
     Program:              Pack Database
     Author:               J D Allen and H R Lang
     Date Written:         August 1987
     Description:          This program packs/compresses the database.
     Tables Used:          none
     Temp Tables:          none
     Reports Used:         none
     Forms Used:           none

*(***********************************************************************
     Program:              Backup Database
     Author:               J D Allen and H R Lang
     Date Written:         August 1987
     Description:          This program provides instructions for
                           backing up the database.
     Tables Used:          none
     Tcmp Tables:          none
     Reports Used:         none
     Forms Used:           none

*(***********************************************************************
     Program:              Restore Database
     Author:               J D Allen and H R Lang
     Date Written:         August 1987
     Description:          This program provides instructions for
                           restoring the database from a backup.
     Tables Used:          none
     Temp Tables:          none
     Reports Used:         none
     Forms Used:           none

************************************************************************)
```

APPENDIX E
SOURCE CODE FOR ANCILLARY PROGRAM MODULES

The Search and Rescue and Enforcement of Laws and Treaties Intelligence System (SARELTIS) Prototype for US Coast Guard Group Monterey consists of eleven sets of application programs written in the R:base 5000 command language, and sixteen ancillary program modules written in either the R:base 5000 command language or as DOS batch or data files. All of these programs have access to one common relational database structure, named SARSYS.

This appendix contains the source code for the sixteen ancillary program modules of SARELTIS. These modules include the following:

- RESTBRD.cmd - Runs the Boarding Reports Menu Program
- RESTINPT.cmd - Runs the Data Entry Menu Program
- RESTINT.cmd - Runs the Intelligence Reports Menu Program
- RESTMAIN.cmd - Runs the Main Menu Program
- RESTMOD.cmd - Runs the Data Modification Menu Program
- RESTPRS.cmd - Runs the Personnel Reports Menu Program
- RESTQRY.cmd - Runs the Query Menu Program
- RESTRPTS.cmd - Runs the Reports Master Menu Program
- RESTSAR.cmd - Runs the Sar Reports Menu Program
- RESTUTLY.cmd - Runs the Utilities Menu Program
- RESTVSL.cmd - Runs the Vessel Reports Menu Program
- STARTUP.BIN - Runs the Title Page and Security Program
- RBASE.DAT - Runs Startup.bin when R:base is loaded
- SARSYS.BAT - Automatically loads application from main directory
- BACKDBMS.BAT - Batch file to backup the database and programs
- RESTDBMS.BAT - Batch file to restore the database and programs

The source code for the application program sets is contained in Appendix F through Appendix P. All programs for this project were developed by the writers. Internal formats for the Reports and Entry/Edit forms are contained within the R:base 5000 database structure itself and were considered inappropriate for listing.

```
*(*********************************************************************

    Program:           RESTBRD.cmd
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program calls the boarding reports menu.
    Tables Used:       none
    Temp Tables:       none
    Reports Used:      none
    Forms Used:        none

**********************************************************************)

run brd_main in brdrpts.apx
*( end )


*(*********************************************************************

    Program:           RESTINPT.cmd
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program calls the input main menu.
    Tables Used:       none
    Temp Tables:       none
    Reports Used:      none
    Forms Used:        none

**********************************************************************)

run inpmain in sentry.apx
*( end )


*(*********************************************************************

    Program:           RESTINT.cmd
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program calls the intelligence reports menu.
    Tables Used:       none
    Temp Tables:       none
    Reports Used:      none
    Forms Used:        none

**********************************************************************)

run int_main in intrpts.apx
*( end )


*(*********************************************************************

    Program:           RESTMAIN.cmd
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program calls the main menu of the system.
    Tables Used:       none
    Temp Tables:       none
    Reports Used:      none
    Forms Used:        none

**********************************************************************)

run sarsys in sarsys.apx
*( end )
```

```
*(**********************************************************************

    Program:            RESTMOD.cmd
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This program calls the modify main menu.
    Tables Used:        none
    Temp Tables:        none
    Reports Used:       none
    Forms Used:         none

***********************************************************************)

run modmain in modify.apx
*( end )


*(*********************************************************************

    Program:            RESTPRS.cmd
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This program calls the personnel reports menu.
    Tables Used:        none
    Temp Tables:        none
    Reports Used:       none
    Forms Used:         none

***********************************************************************)

run prs_main in prsrpts.apx
*( end )


*(**********************************************************************

    Program:            RESTQRY.cmd
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This program calls the modify main menu.
    Tables Used:        none
    Temp Tables:        none
    Reports Used:       none
    Forms Used:         none

***********************************************************************)

run qrymain in squery.apx
*( end )


*(**********************************************************************

    Program:            RESTRPTS.cmd
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This program calls the reports main menu.
    Tables Used:        none
    Temp Tables:        none
    Reports Used:       none
    Forms Used:         none

***********************************************************************)

run rptmain in sreports.apx
```

```
*( end )


*(**********************************************************************

    Program:           RESTSAR.cmd
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program calls the sar reports menu.
    Tables Used:       none
    Temp Tables:       none
    Reports Used:      none
    Forms Used:        none

**********************************************************************)

run sar_main in sarrpts.apx
*( end )


*(**********************************************************************

    Program:           RESTUTLY.cmd
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program calls the utility main menu.
    Tables Used:       none
    Temp Tables:       none
    Reports Used:      none
    Forms Used:        none

**********************************************************************)

run utlmain in sutility.apx
*( end )


*(**********************************************************************

    Program:           RESTVSL.cmd
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program calls the vessel reports menu.
    Tables Used:       none
    Temp Tables:       none
    Reports Used:      none
    Forms Used:        none

**********************************************************************)

run vsl_main in vslrpts.apx
*( end )


*(**********************************************************************

    Program:           STARTUP.prg (uncompiled version of STARTUP.BIN)
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program displays the title screen and
                       verifies user access.
    Tables Used:       none
    Temp Tables:       none
    Reports Used:      none
    Forms Used:        none
***********************************************************************)
set error messages off
set escape off
set messages off
newpage
clear all variables
write "                             WELCOME                             +
 " at 3 5
write "                           This is the                           +
 " at 5 5
write "                        Search & Rescue /                        +
 " at 8 5
write "                Enforcement of Laws and Treaties                 +
 " at 9 5
write "           Intelligence Information System (SARELTIS)            +
 " at 10 5
write "            Authors: LT Jon D. Allen,  US Coast Guard            +
 " at 14 5
write "                     LT Heidi R. Lang,  US Navy                  +
 " at 15 5
write "               Date: September 1987                              +
 " at 16 5
write "                Please press any key to continue                 +
 " at 22 5
pause
newpage
set var username text
set var userpass text
set var validusr text
set var validusr to no
set var chances integer
set var chances to 3
set error messages on
while validusr eq no then
   write "Please enter your user name and your user password" at 3 1
   fillin username using "User Name: " at 5 20
   fillin userpass using "Password: " at 6 21
   set error messages off
   newpage
   if username eq "*****" and userpass eq "******" then
      set var validusr to yes
      break
   endif
   if username eq "***" and userpass eq "******" then
      set var validusr to yes
      break
   endif
   set var chances to .chances - 1
   if chances le 0 then
      newpage
      set var count to 20
      while count ne 0 then
         write "Improper LOGON attempt....SARELTIS Terminated!" at 10 1
         set var count to .count - 1
      endwhile
      newpage
      exit
   endif
   write "Invalid user name or password,  Please try again." at 10 1
endwhile
if username eq sysop then
   user sarsys
   newpage
   write "Initializing SARELTIS" at 10 25
   run sarsys in sarsys.apx
endif
if username eq ood then
   newpage
   write "Initializing SARELTIS" at 10 25
   run sarsys in sarsys.apx
endif
exit
*( end )


*(***************************************************************************

    Program:            RBASE.DAT
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This program loads the title screen and security
                        module when R:base 5000 is loaded.
    Tables Used:        none
    Temp Tables:        none
    Reports Used:       none
    Forms Used:         none

***************************************************************************)

run startup.bin
exit
*( end )


*(***************************************************************************

    Program:            SARSYS.BAT
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This file loads R:base and enters the
                        SARELTIS application directly from the
                        main directory.
    Tables Used:        none
    Temp Tables:        none
    Reports Used:       none
    Forms Used:         none

***************************************************************************)

cd c:\dbdir
path = c:\r5k
rbase
*( end )


*(***************************************************************************

    Program:            BACKDBMS.BAT
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This file performs a backup of the
                        database structure and all application
                        programs.
    Tables Used:        none
    Temp Tables:        none
    Reports Used:       none
    Forms Used:         none

***************************************************************************)

path = c: ;c:\util
cd \dbdir
backup *.apx a:
backup sarsys?.rbs a:
backup *.cmd a:
backup startup.bin
backup rbase.dat
path = c:\dbdir;c:\r5k
rbase
*( end )


*(*********************************************************************

    Program:            RESTDBMS.BAT
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This file restores a backup copy of
                        the database structure and application
                        programs.
    Tables Used:        none
    Temp Tables:        none
    Reports Used:       none
    Forms Used:         none

**********************************************************************)

path = c:\dbdir;c:\util
cd \dbdir
restore a: \dbdir
restore a: \dbdir
restore a: \dbdir
restore a: \dbdir
restore a: \dbdir
path = c:\dbdir;c:\r5k
rbase
*( end )
```

APPENDIX F
SARSYS SOURCE CODE

```
$COMMAND
SARSYS
*(**********************************************************************

                US COAST GUARD GROUP MONTEREY
      SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                INTELLIGENCE SYSTEM (SARELTIS)

              MAIN MENU and STARTUP APPLICATION

             Author:    Jon D. Allen, LT, USCG
                        Heidi R. Lang, LT, USN
                        AUGUST 1987

**********************************************************************)

set escape off
set messages off
set error messages off
open sarsys
clear all variables
run INITIAL in sutility.apx
set var pick1 integer
newpage
choose pick1 from mainmenu in sarsys.apx
if pick1 eq 0 then
   quit to restmain.cmd
endif
if pick1 eq 1 then
   run SINPUTS in sarsys.apx
   quit to restmain.cmd
endif
if pick1 eq 2 then
   run MODIFY in sarsys.apx
   quit to restmain.cmd
endif
if pick1 eq 3 then
   run QUERY in sarsys.apx
   quit to restmain.cmd
endif
if pick1 eq 4 then
   run SREPORTS in sarsys.apx
   quit to restmain.cmd
endif
if pick1 eq 5 then
   run UTILITY in sarsys.apx
   quit to restmain.cmd
endif
if pick1 eq 6 then
   run ADHOC in sarsys.apx
   quit to restmain.cmd
endif
if pick1 eq 7 then
   set var exitprg text
   set var exitprg to yes
endif
write "You have completed this session...." at 10 1
write "REMEMBER --- Do frequent backups!" at 12 1
write "Please press any key to exit" at 18 1
pause
```

```
newpage
set escape on
set messages on
set error messages on
exit
$menu
mainmenu
column SARELTIS --- MASTER MENU
INPUT INFORMATION
MODIFY INFORMATION
QUERY INFORMATION
REPORTS
UTILITIES
AD HOC QUERIES
EXIT SARELTIS
*( end )


$COMMAND
SINPUTS
*(**********************************************************************

    Program:          Access Input Module
    Author:           J D Allen and H R Lang
    Date Written:     August 1987
    Description:      This program calls the input master menu.
    Tables Used:      none
    Temp Tables:      none
    Reports Used:     none
    Forms Used:       none

***********************************************************************)

run INPMAIN in SENTRY.apx
return
*( end )


$COMMAND
MODIFY
*(**********************************************************************

    Program:          Access Modify Module
    Author:           J D Allen and H R Lang
    Date Written:     August 1987
    Description:      This program calls the modify master menu.
    Tables Used:      none
    Temp Tables:      none
    Reports Used:     none
    Forms Used:       none

***********************************************************************)

run MODMAIN in MODIFY.apx
return
*( end )


$COMMAND
QUERY
*(**********************************************************************

    Program:          Access Query Module
    Author:           J D Allen and H R Lang
    Date Written:     August 1987
    Description:      This program calls the query master menu.
    Tables Used:      none
    Temp Tables:      none
    Reports Used:     none
    Forms Used:       none

***********************************************************************)

run QRYMAIN in SQUERY.apx
return
*( end )


$COMMAND
SREPORTS
*(**********************************************************************

    Program:          Access Reports Module
    Author:           J D Allen and H R Lang
    Date Written:     August 1987
    Description:      This program calls the reports master menu.
    Tables Used:      none
    Temp Tables:      none
    Reports Used:     none
    Forms Used:       none

***********************************************************************)

run RPTMAIN in SREPORTS.apx
return
*( end )


$COMMAND
UTILITY
*(**********************************************************************

    Program:          Access Utilities Module
    Author:           J D Allen and H R Lang
    Date Written:     August 1987
    Description:      This program calls the utility master menu.
    Tables Used:      none
    Temp Tables:      none
    Reports Used:     none
    Forms Used:       none

***********************************************************************)

run UTLMAIN in SUTILITY.apx
return
*( end )


$COMMAND
ADHOC
*(**********************************************************************

    Program:          Access Ad Hoc Module
    Author:           J D Allen and H R Lang
    Date Written:     August 1987
    Description:      This program allows the user to exit the
                      menus to the RBase 5000 command prompt to
                      execute ad hoc queries.
    Tables Used:      none
    Temp Tables:      none
    Reports Used:     none
    Forms Used:       none
```

```
**********************************************************************)

newpage
write "You are now exiting the menu system of SARELTIS. Before +
exercising " at 4 1
write "this option,  you should become very familiar with RBase 5000 " +
   at 5 1
write "query commands found in the reference manuals and the actual " +
   at 6 1
write "database structure."  at 7 1
write "Do you wish to return to the SARELTIS menus? (Y)" at 10 1
set var choice text
fillin choice using "(" at 10 46
if choice eq n then
   newpage
   write "You are now exitting the SARELTIS menu system" at 8 10
   write "If you wish to return to the menu system, please" at 10 10
   write "type   RUN STARTUP    at the RBase 5000 prompt." at 11 10
   write "Please press any key to continue" at 16 10
   pause
   set escape on
   set messages on
   set error messages on
   clear all variables
   set bell on
   newpage
   quit
endif
return
*( end )
```

APPENDIX G

SENTRY SOURCE CODE

```
$COMMAND
INPMAIN
*(************************************************************************

                     US COAST GUARD GROUP MONTEREY
         SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                     INTELLIGENCE SYSTEM (SARELTIS)

                  INPUT MASTER MENU and INPUT PROGRAMS

                 Author:    Jon D. Allen, LT, USCG
                            Heidi R. Lang, LT, USN
                            AUGUST 1987

*************************************************************************)

newpage
set messages off
set escape off
clear all variables
set var pick2 integer
newpage
choose pick2 from inp_menu in sentry.apx
if pick2 eq 0 then
   quit to restinpt.cmd
endif
if pick2 eq 1 then
   run add_sar in sentry.apx
   quit to restinpt.cmd
endif
if pick2 eq 2 then
   run add_sbd in sentry.apx
   quit to restinpt.cmd
endif
if pick2 eq 3 then
   run add_sut in sentry.apx
   quit to restinpt.cmd
endif
if pick2 eq 4 then
   run add_brd in sentry.apx
   quit to restinpt.cmd
endif
if pick2 eq 5 then
   run add_ownr in sentry.apx
   quit to restinpt.cmd
endif
if pick2 eq 6 then
   run am_vint in modify.apx
   quit to restinpt.cmd
endif
if pick2 eq 7 then
   run am_pint in modify.apx
   quit to restinpt.cmd
endif
if pick2 eq 8 then
   quit to restmain.cmd
endif
quit to restinpt.cmd
return
```

```
$menu
inp_menu
column SARELTIS --- INPUT MENU
INPUT SAR INFORMATION (NO BOARDING)
INPUT SAR INFORMATION WITH BOARDING
INPUT MULTIPLE SAR UNITS
INPUT BOARDING INFORMATION
INPUT MULTIPLE VESSEL OWNERS
INPUT VESSEL INTELLIGENCE
INPUT PERSONNEL INTELLIGENCE
RETURN TO MAIN MENU
*( end )


$COMMAND
ADD_SAR
*(*********************************************************************

    Program:            Add Sar
    Author:             H R Lang and J D Allen
    Date Written:       July 1987
    Description:        This program allows the user to input
                        sar case information.
    Tables Used:        vessel, sar, owners, operators, owvsl, opvsl
    Temp Tables:        none
    Reports Used:       none
    Forms Used:         sarform1, sarform2, sarform3, sarform4,
                        sarform5, sarform6, sarform7, sarform8,
                        sarform9

*********************************************************************)

set error messages on
set messages off
set escape off
newpage
write "Program loading in progress.  Please stand by ........" at 10 10
clear all variables
set null ?
set var pg to 1
*(initialize edit and load control variables)
set var review to N
set var loadmenu to N
label begin
set var backup to N
set var pgcount to 0
while pg gt 0 then  *(main loop)
  newpage
  if pg eq 1 then
    *(enter general sar information)
    if review eq N and backup eq N then
      set var pgcount to 1
      set var vsardate to .#date
      set var vsartime to 12:00:00
      set var vsarlat to 0.0
      set var vsarlong to 0.0
      set var vdistoff to 0
      set var vsarpob to 0
    endif
    write "Press [ESC] to quit or [PGDN] to move down a page" at 1 1
    draw sarform1 with all at 3
    if review eq N then
      edit var return esc pgdn
    else
      edit var vfoldno vucn vmucn vsardate vsarday vsartime vsarnod +
      vimmed vsarlat vsarlong vdistoff vsarpob return esc pgdn
    endif
  endif
```

```
if pg eq 2 then
  *(enter vessel information)
  if review eq N and backup eq N then
    set var pgcount to 2
    while vvslhin exists then
      set pointer #1 e1 for vessel where vslhin = .vvslhin
      newpage
      if e1 eq 0 then
        *(vessel exists in database)
        set var vvslno to vslno in #1
        set var vvslname to vslname in #1
        set var vvslmake to vslmake in #1
        set var vvslmod to vslmodel in #1
        set var vvslyear to vslyear in #1
        set var vvslton to vsltons in #1
        set var vvslft to vsllenft in #1
        set var vvslin to vsllenin in #1
        set var vvslhp to vslhp in #1
        set var vvsluse to vsluse in #1
        set var vvsltype to vsltype in #1
        set var vvslprop to vslprop in #1
        set var vvslhull to vslhull in #1
        set var vvsleng to vsleng in #1
        set var vvslfuel to vslfuel in #1
        set var vconstr to vslconst in #1
        set var vdecal to cmedecal in #1
        set var vcmeyear to cmeyear in #1
        set var voffno to vsloffno in #1
        set var vhome to homeport in #1
        set var vcall to vslcall in #1
        set var vvalue to vslvalue in #1
        set var vsuper to vslsuper in #1
        set var vmasts to vslmasts in #1
        set var vsailno to sailno in #1
        set var vhlcolor to hlcolor in #1
        set var vtrcolor to trcolor in #1
        set var vslcolor to slcolor in #1
        set var vcbcolor to cbcolor in #1
        set var vdkcolor to dkcolor in #1
        set var vmarks to vslmarks in #1
        set var v1 to vslno in #1
        set var v2 to vslname in #1
        set var v3 to .slhin in #1
        set var v4 to vslmake in #1
        set var v5 to vslmodel in #1
        set var v6 to vslyear in #1
        set var v7 to vsltons in #1
        set var v8 to vsllenft in #1
        set var v9 to vsllenin in #1
        set var v10 to vslhp in #1
        set var v11 to vsluse in #1
        set var v12 to vsltype in #1
        set var v13 to vslprop in #1
        set var v14 to vslhull in #1
        set var v15 to vsleng in #1
        set var v16 to vslfuel in #1
        set var v17 to vslconst in #1
        set var v18 to cmedecal in #1
        set var v19 to cmeyear in #1
        set var v20 to vsloffno in #1
        set var v21 to homeport in #1
        set var v22 to vslcall in #1
        set var v23 to vslvalue in #1
        set var v24 to vslsuper in #1
        set var v25 to vslmasts in #1
        set var v26 to sailno in #1
        set var v27 to hlcolor in #1
        set var v28 to trcolor in #1
        set var v29 to slcolor in #1
        set var v30 to cbcolor in #1
```

```
          set var v31 to dkcolor in #1
          set var v32 to vslmarks in #1
        else
          set var vvsltype text
          set var vvsltype to 9
          set var vvsluse text
          set var vvsluse to 0
          set var vvslyear text
          set var vvslyear to 1901
          set var vvslhp text
          set var vvslhp to 0
          set var vvslft to 0
          set var vvslin to 0
          set var vvslton to 0
          set var vvalue to $0
          set var vdecal text
          set var vcmeyear text
        endif
        break
      endwhile
    endif
    write "Press [ESC] to quit, [PGUP] to move up, [PGDN] to move +
down a page" at 1 1
    draw sarform2 with all at 3
    edit var vvslno voffno vhome vcall vvsltype vvsluse vvalue +
    vvslmake vvslmod vvslyear vvslft vvslin vvslhp vvslton return +
    esc pgup pgdn
  endif
  if pg eq 3 then
    *(enter vessel information)
    if review eq N and backup eq N then
      set var pgcount to 3
      if el ne 0 then
        set var vvslhull text
        set var vvslhull to 8
        set var vvslfuel text
        set var vvslfuel to 0
        set var vvsleng text
        set var vvsleng to 0
        set var vconstr text
        set var vconstr to 0
        set var vvslprop text
        set var vvslprop to 9
        set var vslcolor to unk
        set var vhlcolor to unk
        set var vtrcolor to unk
        set var vcbcolor to unk
        set var vdkcolor to unk
      endif
    endif
    write "Press [ESC] to quit, [PGUP] to move up, [PGDN] to move +
down a page" at 1 1
    draw sarform3 with all at 3
    edit var vvslhull vvslfuel vvsleng vconstr vvslprop vsuper vmasts +
    vslcolor vsailno vhlcolor vtrcolor vcbcolor vdkcolor vmarks +
    return esc pgup pgdn
  endif
  if pg eq 4 then
    *(enter owner information)
    if review eq N and backup eq N then
      set var pgcount to 4
      set var vowlname to unk
      set var vowfname to unk
      write "Press [PGDN] after entering initial information to check
if" at 1 5
      write "owner exists in database.  Press [ESC] to quit" at 2 10
      draw sarform4 with all at 4
      edit var return esc pgdn
    endif
    if #return ne esc then
```

```
   if review eq N and backup eq N then
    while vowlname exists and vowfname exists then
     set pointer #2 e2 for owners where ownlname = .vowlname and +
     ownfname = .vowfname
     newpage
     while e2 eq 0 then
      *(owner last/first name exists in database)
      set var vowmi to ownmi in #2
      set var vowdob to owndob in #2
      set var vowaddr to ownaddr in #2
      set var vowcity to owncity in #2
      set var vowst to ownstate in #2
      set var vowzip to ownzip in #2
      set var vowtelno to owntelno in #2
      newpage
      draw sarform4 with all at 3
      draw sarform5 with all at 8
      write "Correct owner? (N)" at 21 20
      fillin owfound using "(" at 21 35
      if owfound exists and owfound eq Y then
       set var ow1 to .vowlname
       set var ow2 to .vowfname
       set var ow3 to ownmi in #2
       set var ow4 to ownaddr in #2
       set var ow5 to owncity in #2
       set var ow6 to ownstate in #2
       set var ow7 to ownzip in #2
       set var ow8 to owntelno in #2
       set var ow9 to owndob in #2
       set var ow10 to ownid in #2
       set var vownid to ownid in #2
       break
      endif
      next #2 e2
     endwhile .
     if e2 ne 0 then
      newpage
      write "This owner does not exist in the database" at 5 5
      fillin go using "Press [ENTER] to continue" at 9 5
      clear vowmi
      clear vowaddr
      clear vowcity
      set var vowst to CA
      clear vowzip
      clear vowtelno
      set var vowdob to 01/01/1901
      compute vmaxown as max ownid from owners
      if vmaxown fails then
       set var vmaxown to 0
      endif
      set var vownid to .vmaxown + 1
     endif
     break
    endwhile
   endif
   newpage
   write "Press [ESC] to quit, [PGUP] to move up, [PGDN] to move +
down a page" at 1 1
   draw sarform4 with all at 3
   draw sarform5 with all at 8
   edit var return esc pgup pgdn
  endif
 endif
 if pg eq 5 then
  *(enter operator information)
  if review eq N and backup eq N then
   set var pgcount to 5
   write "Press [PGDN] after entering initial information to check
if" at 1 5
   write "operator exists in database.  Press [ESC] to quit" at 2 10
  set var voplname to .vowlname
  set var vopfname to .vowfname
  draw sarform6 with all at 4
  edit var return esc pgdn
endif
if #return ne esc then
  if review eq N and backup eq N then
    while voplname exists and vopfname exists then
      set pointer #3 e3 for operator where oplname = .voplname +
      and opfname = .vopfname
      newpage
      while e3 eq 0 then
        *(operator last/first name exists in database)
        set var vopmi to opmi in #3
        set var vopdob to opdob in #3
        set var vopaddr to opaddr in #3
        set var vopcity to opcity in #3
        set var vopst to opstate in #3
        set var vopzip to opzip in #3
        set var voptelno to optelno in #3
        set var vopcrse to opcourse in #3
        newpage
        draw sarform6 with all at 3
        draw sarform7 with all at 8
        write "Correct operator? (N)" at 21 20
        fillin opfound using "(" at 21 38
        if opfound exists and opfound eq Y then
          set var op1 to .voplname
          set var op2 to .vopfname
          set var op3 to opmi in #3
          set var op4 to opaddr in #3
          set var op5 to opcity in #3
          set var op6 to opstate in #3
          set var op7 to opzip in #3
          set var op8 to optelno in #3
          set var op9 to opdob in #3
          set var op10 to opcourse in #3
          set var op11 to opid in #3
          set var vopid to opid in #3
          break
        endif
        next #3 e3
      endwhile
      if e3 ne 0 then
        newpage
        write "This operator does not exist in the database" at 5 5
        fillin go using "Press [ENTER] to continue" at 9 5
        if vowlname eq .voplname and vowfname eq .vopfname then
          set var vopmi to .vowmi
          set var vopaddr to .vowaddr
          set var vopcity to .vowcity
          set var vopst to .vowst
          set var vopzip text
          set var vopzip to .vowzip
          set var voptelno text
          set var voptelno to .vowtelno
          set var vopdob to .vowdob
        else
          clear vopmi
          clear vopaddr
          clear vopcity
          set var vopst to CA
          clear vopzip
          clear voptelno
          set var vopdob to 01/01/1901
        endif
        clear vopcrse
        compute vmaxop as max opid from operator
        if vmaxop fails then
          set var vmaxop to 0
```

```
            endif
            set var vopid to .vmaxop + 1
          endif
          break
        endwhile
      endif
      newpage
      write "Press [ESC] to quit, [PGUP] to move up, [PGDN] to move +
down a page" at 1 1
      draw sarform6 with all at 3
      draw sarform7 with all at 8
      edit var return esc pgup pgdn
    endif
  endif
  if pg eq 6 then
    *(enter sar weather information)
    if review eq N and backup eq N then
      set var pgcount to 6
      set var vsunrise to 06:00:00
      set var vsunset to 19:00:00
      set var vceiling to 0
      set var vclouds to 0
      set var vvisdist to 0
      set var vwinddir to 0
      set var vwindspd to 0
      set var vseas to 0
      set var vswells to 0
    endif
    write "Press [ESC] to quit or [PGUP] to move up a page" at 1 1
    draw sarform9 with all at 3
    edit var vsunrise vsunset vceiling vclouds vvisdist vwinddir +
vwindspd vseas vswells return esc pgup
    set var loadmenu to "Y"
  endif
  if #return eq pgup then
    set var backup to Y
    if pg gt 1 then
      set var pg to .pg - 1
    else
      set var pg to 1
    endif
  endif
  if #return eq pgdn then
    if pg lt 6 then
      set var pg to .pg + 1
      if pgcount lt .pg then
        set var backup to N
      endif
    else
      set var pg to 6
    endif
  endif
  if #return eq esc then
    set var backup to N
    set var pg to 0
  endif
endwhile  *(end of main loop)
newpage
write "Program loading in progress.  Please stand by ........" at 10 10
if loadmenu eq "Y" then
  set var choice to 0
  while choice lt 1 or choice gt 3 then
    newpage
    write "Data Load Options" at 2 20
    write "-----------------" at 3 20
    write "Enter:" at 5 20
    write "1-To load the data into the database" at 6 20
    write "2-To ignore the data entered" at 7 20
    write "3-To go back and verify data entered" at 8 20
    write "Your choice: " at 10 20
```

```
fillin choice using " " at 10 33
newpage
if choice eq 1 then
  *(test for required information)
  if vsardate exists and vfoldno exists and vucn exists and +
  vvslname exists and vsarlat exists and vsarlong exists and +
  vsarnod exists and vimmed exists then
    set error messages off
    if e1 eq 0 then
      *(vessel exists in database)
      write "changing values in vessel"
      delete rows from #1
    else
      write "loading vessel"
    endif
    load vessel
      .vvslno .vvslname .vvslhin .vvslmake .vvslmod .vvslyear +
      .vvslton .vvslft .vvslin .vvslhp .vvsluse .vvsltype +
      .vvslprop .vvslhull .vvsleng .vvslfuel .vconstr .vdecal +
      .vcmeyear .voffno .vhome .vcall .vvalue .vsuper .vmasts +
      .vsailno .vhlcolor .vtrcolor .vslcolor .vcbcolor .vdkcolor +
      .vmarks
    end
    if e1 eq 0 then
      set pointer #1 e1 for vessel where vslhin eq .v3
      if e1 ne 0 then
        load vessel
          .v1 .v2 .v3 .v4 .v5 .v6 .v7 .v8 .v9 .v10 .v11 .v12 .v13 +
          .v14 .v15 .v16 .v17 .v18 .v19 .v20 .v21 .v22 .v23 .v24 +
          .v25 .v26 .v27 .v28 .v29 .v30 .v31 .v32
        end
      endif
    endif
    set pointer #1 e1 for sar where folderno = .foldno
    if e1 eq 0 then
      set var difsar to "F"
      set var s1 to sardate in #1
      set var s2 to sartime in #1
      set var s3 to sarlat in #1
      set var s4 to sarlong in #1
      set var s5 to sarnod in #1
      set var s6 to sarvhin in #1
      if s1 ne .vsardate or s2 ne .vsartime or s3 ne .vsarlat or +
      s4 ne .vsarlong or s5 ne .vsarnod or s6 ne .vvslhin then
        set var difsar to "T"
      endif
    else
      set var difsar to "T"
    endif
    if difsar eq T then
      write "loading sar"
      load sar
        .vfoldno .vucn .vmucn .vsardate .vsarday .vsartime +
        .vsunrise .vsunset .vsarnod .vsarpob .vvslhin .vsarlat +
        .vsarlong .vdistoff .vimmed .vwinddir .vwindspd .vvisdist +
        .vclouds .vceiling .vseas .vswells
      end
    endif
    write "loading owvsl"
    load owvsl
      .vvslhin .vownid
    end
    write "loading opvsl"
    load opvsl
      .vvslhin .vopid
    end
    if e2 eq 0 then
      *(owner exists in database)
      write "changing values in owner"
      delete rows from #2
  else
    write "loading owner"
  endif
  load owners
    .vowlname .vowfname .vowmi .vowaddr .vowcity .vowst .vowzip +
    .vowtelno .vowdob .vownid
  end
  if e2 eq 0 then
    set pointer #2 e2 for owners whe ownid eq .ow10
    if e2 ne 0 then
      load owners
        .ow1 .ow2 .ow3 .ow4 .ow5 .ow6 .ow7 .ow8 .ow9 .ow10
      end
    endif
  endif
  if e3 eq 0 then
    *(operator exists in database)
    write "changing values in operator"
    delete rows from #3
  else
    write "loading operator"
    set var vopcrse text
  endif
  load operator
    .voplname .vopfname .vopmi .vopaddr .vopcity .vopst .vopzip +
    .voptelno .vopdob .vopcrse .vopid
  end
  if e3 eq 0 then
    set pointer #3 e3 for operator where opid eq .op11
    if e3 ne 0 then
      load operator
        .op1 .op2 .op2 .op3 .op4 .op5 .op6 .op7 .op8 .op9 .op10 +
        .op11
      end
    endif
  endif
  set error messages on
  write "   "
  write "Return to correct data entered? (Y)" at 20 10
  write "Note:  correct only errors indicated by messages" +
  at 22 5
  fillin er using "(" at 20 42
  if er fails or er eq "Y" then
    set var review to "Y"
    set var pg to 1
    set pointer #1 e1 for vessel where vslhin eq .vvslhin
    set pointer #2 e2 for owners where ownid eq .vownid
    set pointer #3 e3 for operator where opid eq .vopid
    break
  else
    newpage
    write "Removing duplicate entries from tables" at 10 10
    write "Please stand by ................" at 15 12
    delete duplicates from vessel
    delete duplicates from owvsl
    delete duplicates from opvsl
    delete duplicates from owners
    delete duplicates from operator
  endif
else
  newpage
  write "Load not performed.  Required information missing." +
  at 5 20
  write "Required information includes:" at 10 30
  write "Sar Date, Folder Number, Unit Case Number" at 12 35
  write "Vessel Name, Sar Latitude, Sar Longitude" at 13 35
  write "Sar Nature of Distress" at 14 35
  fillin go using "Press [ENTER] to return to data entry" +
  at 16 25
  set var pg to 1
```

```
    set var review to "Y"
    break
  endif
endif
if choice eq 1 or choice eq 2 then
  *(enter sar units)
  newpage
  write "Enter sar units? (Y)" at 12 20
  fillin s_units using "(" at 12 37
  if s_units fails or s_units eq Y then
    set var more to "Y"
    while more eq Y then
      set var vuwtime to .vsartime
      set var vuwdate to .vsardate
      set var vetadate to .vsardate
      set var vdateos to .vsardate
      set var valgdate to .vsardate
      set var vtowdate to .vsardate
      set var vdvmoord to .vsardate
      set var vsumoord to .vsardate
      newpage
      write "Press [PGDN] to load data or [ESC] to quit" at 1 1
      draw sarform8 with all at 3
      edit var vunit vuwtime vuwdate voseta vetadate vostime +
      vdateos valgtime valgdate vtowtime vtowdate vdvmoort +
      vdvmoord vsumoort vsumoord return esc pgdn
      if #return eq pgdn then
        if voseta fails then
          clear vetadate
          set var vetadate date
        endif
        if vostime fails then
          clear vdateos
          set var vdateos date
        endif
        if valgtime fails then
          clear valgdate
          set var valgdate date
        endif
        if vtowtime fails then
          clear vtowdate
          set var vtowdate date
        endif
        if vdvmoort fails then
          clear vdvmoord
          set var vdvmoord date
        endif
        if vsumoort fails then
          clear vsumoord
          set var vsumoord date
        endif
        set pointer #1 e1 for sarunits whe foldno eq .vfoldno and +
        unitname eq .vunit
        newpage
        if e1 eq 0 then
          write "Duplicate entry not loaded"
        else
          write "Loading sarunits"
          load sarunits
            .vfoldno .vunit .vuwtime .vuwdate .voseta .vetadate +
            .vostime .vdateos .valgtime .valgdate .vtowtime +
            .vtowdate .vdvmoort .vdvmoord .vsumoort .vsumoord
          end
        endif
        write "More units to enter? (N)" at 23 20
        fillin mrunits using "(" at 23 41
        if mrunits fails or mrunits eq N then
          set var more to N
        else
          clear vunit
```

```
              clear voseta
              clear vostime
              clear valgtime
              clear vtowtime
              clear vdvmoort
              clear vsumoort
            endif
          endif
          if #return eq esc then
            set var more to N
          endif
        endwhile
      endif
      *(enter owner information)
      newpage
      write "Enter additional owners? (N)" at 12 20
      fillin addlownr using "(" at 12 45
      if addlownr exists and addlownr eq Y then
        set var more to "Y"
        while more eq Y then
          newpage
          set var vowlname to unk
          set var vowfname to unk
          write "Press [PGDN] after entering initial information to +
check if" at 1 5
          write "owner exists in database.  Press [ESC] to quit" +
          at 2 10
          draw sarform4 with all at 4
          edit var return esc pgdn
          if #return ne esc then
            while vowlname exists and vowfname exists then
              set pointer #2 e2 for owners where ownlname = .vowlname +
              and ownfname = .vowfname
              newpage
              while e2 eq 0 then
                *(owner last/first name exists in database)
                set var vowmi to ownmi in #2
                set var vowdob to owndob in #2
                set var vowaddr to ownaddr in #2
                set var vowcity to owncity in #2
                set var vowst to ownstate in #2
                set var vowzip to ownzip in #2
                set var vowtelno to owntelno in #2
                newpage
                draw sarform4 with all at 3
                draw sarform5 with all at 8
                write "Correct owner? (N)" at 21 20
                fillin owfound using "(" at 21 35
                if owfound exists and owfound eq Y then
                  set var ow1 to .vowlname
                  set var ow2 to .vowfname
                  set var ow3 to ownmi in #2
                  set var ow4 to ownaddr in #2
                  set var ow5 to owncity in #2
                  set var ow6 to ownstate in #2
                  set var ow7 to ownzip in #2
                  set var ow8 to owntelno in #2
                  set var ow9 to owndob in #2
                  set var ow10 to ownid in #2
                  set var vownid to ownid in #2
                  break
                endif
                next #2 e2
              endwhile
              if e2 ne 0 then
                newpage
                write "This owner does not exist in the database" +
                at 5 5
                fillin go using "Press [ENTER] to continue" at 9 5
                clear vowmi
                clear vowaddr
                clear vowcity
                set var vowst to CA
                clear vowzip
                clear vowtelno
                set var vowdob to 01/01/1901
                compute vmaxown as max ownid from owners
                if vmaxown fails then
                  set var vmaxown to 0
                endif
                set var vownid to .vmaxown + 1
              endif
              break
            endwhile
            label redoown
            newpage
            write "Press [PGDN] to load data or [ESC] to quit" at 1 1
            draw sarform4 with all at 3
            draw sarform5 with all at 8
            edit var return esc pgup pgdn
            if #return eq pgdn then
              newpage
              set error messages off
              if e2 eq 0 then
                *(owner exists in database)
                write "changing values in owner"
                delete rows from #2
              else
                write "loading owner"
              endif
              load owners
                .vowlname .vowfname .vowmi .vowaddr .vowcity .vowst +
                .vowzip .vowtelno .vowdob .vownid
              end
              if e2 eq 0 then
                set pointer #2 e2 for owners where ownid eq .ow10
                if e2 ne 0 then
                  load owners
                    .ow1 .ow2 .ow3 .ow4 .ow5 .ow6 .ow7 .ow8 .ow9 .ow10
                  end
                endif
              endif
              set error messages on
              write "Return to correct errors? (N)" at 15 10
              write "Note:  correct only errors indicated by +
messages" at 20 5
              fillin fixown using "(" at 15 36
              if fixown exists and fixown eq Y then
                goto redoown
              endif
              set pointer #2 e2 for owvsl where vhinown eq .vvslhin +
              and idowner eq .vownid
              *(check if record already exists)
              newpage
              if e2 ne 0 then
                load owvsl
                  .vvslhin .vownid
                end
              endif
            endif
          endif
          write "More owners to enter? (N)" at 20 20
          fillin mrownrs using "(" at 20 42
          if mrownrs fails or mrownrs eq N then
            set var more to N
          endif
        endwhile
      endif
      newpage
      write "Enter additional operators? (N)" at 12 20
      fillin addloper using "(" at 12 48
      if addloper exists and addloper eq Y then
        set var more to "Y"
        while more eq Y then
          newpage
          *(enter operator information)
          write "Press [PGDN] after entering initial information to +
check if" at 1 5
          write "operator exists in database.  Press [ESC] to quit" +
          at 2 10
          set var voplname to unk
          set var vopfname to unk
          draw sarform6 with all at 4
          edit var return esc pgdn
          if #return ne esc then
            while voplname exists and vopfname exists then
              set pointer #3 e3 for operator where oplname = +
              .voplname and opfname = .vopfname
              newpage
              while e3 eq 0 then
                *(operator last/first name exists in database)
                set var vopmi to opmi in #3
                set var vopdob to opdob in #3
                set var vopaddr to opaddr in #3
                set var vopcity to opcity in #3
                set var vopst to opstate in #3
                set var vopzip to opzip in #3
                set var voptelno to optelno in #3
                set var vopid to opid in #3
                set var vopcrse to opcourse in #3
                newpage
                draw sarform6 with all at 3
                draw sarform7 with all at 8
                write "Correct operator? (N)" at 21 20
                fillin opfound using "(" at 21 38
                if opfound exists and opfound eq Y then
                  set var op1 to .voplname
                  set var op2 to .vopfname
                  set var op3 to opmi in #3
                  set var op4 to opaddr in #3
                  set var op5 to opcity in #3
                  set var op6 to opstate in #3
                  set var op7 to opzip in #3
                  set var op8 to optelno in #3
                  set var op9 to opdob in #3
                  set var op10 to opcourse in #3
                  set var op11 to opid in #3
                  break
                endif
                next #3 e3
              endwhile
              if e3 ne 0 then
                newpage
                write "This operator does not exist in the database" +
                at 5 5
                fillin go using "Press [ENTER] to continue" at 9 5
                clear vopmi
                clear vopaddr
                clear vopcity
                set var vopst to CA
                clear vopzip
                clear voptelno
                set var vopdob to 01/01/1901
                clear vopcrse
                compute vmaxop as max opid from operator
                if vmaxop fails then
                  set var vmaxop to 0
                endif
                set var vopid to .vmaxop + 1
              endif
```

```
    break
   endwhile
   label redooper
   newpage
   write "Press [PGDN] to load data or [ESC] to quit" at 1 1
   draw sarform6 with all at 3
   draw sarform7 with all at 8
   edit var return esc pgup pgdn
   if #return eq pgdn then
    newpage
    set error messages off
    if e3 eq 0 then
     *(operator exists in database)
     write "changing values in operator"
     delete rows from #3
    else
     write "loading operator"
     set var vopcrse text
    endif
    load operator
     .voplname .vopfname .vopmi .vopaddr .vopcity .vopst +
     .vopzip .voptelno .vopdob .vopcrse .vopid
    end
    if e3 eq 0 then
     set pointer #3 e3 for operator where opid eq .op11
     if e3 ne 0 then
      load operator
       .op1 .op2 .op2 .op3 .op4 .op5 .op6 .op7 .op8 .op9 +
       .op10 .op11
      end
     endif
    endif
    set error messages on
    write "Return to correct errors? (N)" at 15 10
    write "Note:  correct only errors indicated by messages" +
    at 20 5
    fillin fixopr using "(" at 15 36
    if fixopr exists and fixopr eq Y then
     goto redooper
    endif
    set pointer #3 e3 for opvsl where vhinoper eq .vvslhin +
    and idoper eq .vopid
    *(check if record already exists)
    newpage
    if e3 ne 0 then
     load opvsl
      .vvslhin .vopid
     end
    endif
   endif
  endif
  write "More operators to enter? (N)" at 20 20
  fillin mropers using "(" at 20 45
  if mropers fails or mropers eq N then
   set var more to N
  endif
 endwhile
endif
newpage
write "Enter another sar case? (Y)" at 12 20
fillin cont using "(" at 12 44
if cont fails or cont eq Y then
 clear all variables
 set var pg to 1
 set var review to "N"
 set var loadmenu to "N"
 break
else
 set null -0-
 set var pg to 0
```

```
      endif
    endif
    if choice eq 3 then
      set var review to "Y"
      set var pg to 1
    endif
  endwhile
else
  newpage
  write "Enter another sar case? (Y)" at 12 20
  fillin cont using "(" at 12 44
  if cont fails or cont eq Y then
    clear all variables
    set var pg to 1
    set var review to "N"
    set var loadmenu to "N"
  else
    set error messages off
    set null -0-
    set var pg to 0
  endif
endif
if pg eq 1 then
  goto begin
endif
return
*( end )


$COMMAND
ADD_SBD
*(**********************************************************************

    Program:          Add Sar with Boarding
    Author:           H R Lang and J D Allen
    Date Written:     July 1987
    Description:      This program allows the user to enter sar
                      and boarding information.
    Tables Used:      boarding, vessel, owners, operator, owvsl,
                      opvsl, sar
    Temp Tables:      none
    Reports Used:     none
    Forms Used:       cg4100a1, cg4100a2, cg4100a3, cg4100a4,
                      cg4100a5, cg4100a6, cg4100a7, cg4100a8,
                      cg4100a9, sbdform1, sarform1, sarform8,
                      sarform9, sarform5

**********************************************************************)

set error messages off
set messages off
set escape off
newpage
write "Program loading in progress.  Please stand by ........" at 10 10
clear all variables
set null ?
set var pg to 1
*(initialize edit and load control variables)
set var review to "N"
set var loadmenu to "N"
label begin
set var backup to N
set var pgcount to 0
while pg gt 0 then  *(main loop)
  newpage
  if pg eq 1 then
    *(enter boarding/vessel information)
    if review eq N and backup eq N then
```

```
      set var pgcount to 1
      set var vbodate to .#date
      set var vbotime to 12:00:00
      set var vvslname to unk
      write "Press [PGDN] after entering initial information to check +
if" at 1 5
      write "vessel exists in database.  Press [ESC] to quit" at 2 10
      draw cg4100a1 with all at 4
      edit var return esc pgdn
    else
      write "Press [PGDN] after review/correction or [ESC] to quit" +
      at 1 5
      draw cg4100a1 with all at 4
      edit var vvslname vbodate vbotime vbono return esc pgdn
    endif
    if #return ne esc then
      while vvslhin exists then
        if review eq N and backup eq N then
          set pointer #1 el for vessel where vslhin = .vvslhin
          newpage
          set var vvslpob to 0
          set var vadult to 0
          set var vchild to 0
          if el eq 0 then
            *(vessel exists in database)
            set var vvslno to vslno in #1
            set var vvslname to vslname in #1
            set var vvslmake to vslmake in #1
            set var vvslmod to vslmodel in #1
            set var vvslyear to vslyear in #1
            set var vvslton to vsltons in #1
            set var vvslft to vsllenft in #1
            set var vvslin to vsllenin in #1
            set var vvslhp to vslhp in #1
            set var vvsluse to vsluse in #1
            set var vvsltype to vsltype in #1
            set var vvslprop to vslprop in #1
            set var vvslhull to vslhull in #1
            set var vvsleng to vsleng in #1
            set var vvslfuel to vslfuel in #1
            set var vconstr to vslconst in #1
            set var vdecal to cmedecal in #1
            set var vcmeyear to cmeyear in #1
            set var voffno to vsloffno in #1
            set var vhome to homeport in #1
            set var vcall to vslcall in #1
            set var vvalue to vslvalue in #1
            set var vsuper to vslsuper in #1
            set var vmasts to vslmasts in #1
            set var vsailno to sailno in #1
            set var vhlcolor to hlcolor in #1
            set var vtrcolor to trcolor in #1
            set var vslcolor to slcolor in #1
            set var vcbcolor to cbcolor in #1
            set var vdkcolor to dkcolor in #1
            set var vmarks to vslmarks in #1
            set var v1 to vslno in #1
            set var v2 to vslname in #1
            set var v3 to vslhin in #1
            set var v4 to vslmake in #1
            set var v5 to vslmodel in #1
            set var v6 to vslyear in #1
            set var v7 to vsltons in #1
            set var v8 to vsllenft in #1
            set var v9 to vsllenin in #1
            set var v10 to vslhp in #1
            set var v11 to vsluse in #1
            set var v12 to vsltype in #1
            set var v13 to vslprop in #1
            set var v14 to vslhull in #1
            set var v15 to vsleng in #1
            set var v16 to vslfuel in #1
            set var v17 to vslconst in #1
            set var v18 to cmedecal in #1
            set var v19 to cmeyear in #1
            set var v20 to vsloffno in #1
            set var v21 to homeport in #1
            set var v22 to vslcall in #1
            set var v23 to vslvalue in #1
            set var v24 to vslsuper in #1
            set var v25 to vslmasts in #1
            set var v26 to sailno in #1
            set var v27 to hlcolor in #1
            set var v28 to trcolor in #1
            set var v29 to slcolor in #1
            set var v30 to cbcolor in #1
            set var v31 to dkcolor in #1
            set var v32 to vslmarks in #1
          else
            *(vessel does not exist in database)
            set var vvslmake to unk
            set var vvslmod to unk
            set var vvslyear text
            set var vvslyear to 1901
            set var vvslhp text
            set var vvslhp to 0
            set var vvsluse text
            set var vvsluse to 0
            set var vvsltype text
            set var vvsltype to 9
            set var vvslprop text
            set var vvslprop to 9
            set var vvslhull text
            set var vvslhull to 8
            set var vvsleng text
            set var vvsleng to 0
            set var vvslfuel text
            set var vvslfuel to 0
            set var vconstr text
            set var vconstr to 0
            set var vvslton to 0
            set var vvslft to 0
            set var vvslin to 0
            set var vdecal to N
          endif
        endif
        newpage
        write "Press [ESC] to quit or [PGDN] to move down a page" +
        at 1 1
        draw cg4100a1 with all at 3
        draw cg4100a2 with all at 10
        edit var return esc pgdn
        break
      endwhile
    endif
  endif
  if pg eq 2 then
   *(enter owner information)
    if review eq N and backup eq N then
      set var pgcount to 2
      set var vowlname to unk
      set var vowfname to unk
      write "Press [PGDN] after entering initial information to check +
if" at 1 5
      write "owner exists in database.  Press [ESC] to quit" at 2 10
      draw cg4100a3 with all at 4
      edit var return esc pgdn
    endif
    if #return ne esc then
      if review eq N and backup eq N then
```

```
        while vowlname exists and vowfname exists then
          set pointer #2 e2 for owners where ownlname = .vowlname and +
          ownfname = .vowfname
          newpage
          while e2 eq 0 then
            *(owner last/first name exist in database)
            set var vowmi to ownmi in #2
            set var vowdob to owndob in #2
            set var vowaddr to ownaddr in #2
            set var vowcity to owncity in #2
            set var vowst to ownstate in #2
            set var vowzip to ownzip in #2
            set var vowtelno to owntelno in #2
            newpage
            draw cg4100a3 with all at 3
            draw cg4100a4 with all at 8
            write "Correct owner? (N)" at 21 20
            fillin owfound using "(" at 21 35
            if owfound exists and owfound eq Y then
              set var ow1 to .vowlname
              set var ow2 to .vowfname
              set var ow3 to ownmi in #2
              set var ow4 to ownaddr in #2
              set var ow5 to owncity in #2
              set var ow6 to ownstate in #2
              set var ow7 to ownzip in #2
              set var ow8 to owntelno in #2
              set var ow9 to owndob in #2
              set var ow10 to ownid in #2
              set var vownid to ownid in #2
              break
            endif
            next #2 e2
          endwhile
          if e2 ne 0 then
            newpage
            write "This owner does not exist in the database" at 5 5
            fillin go using "Press [ENTER] to continue" at 9 5
            clear vowmi
            clear vowaddr
            clear vowcity
            set var vowst to CA
            clear vowzip
            clear vowtelno
            set var vowdob to 01/01/1901
            clear vostatus
            compute vmaxown as max ownid from owners
            if vmaxown fails then
              set var vmaxown to 0
            endif
            set var vownid to .vmaxown + 1
          endif
          break
        endwhile
      endif
      newpage
      write "Press [ESC] to quit, [PGUP] to move up, [PGDN] to move +
down a page" at 1 1
      draw cg4100a3 with all at 3
      draw cg4100a4 with all at 8
      edit var return esc pgup pgdn
    endif
  endif
  if pg eq 3 then
   *(enter operator information)
    if review eq N and backup eq N then
      set var pgcount to 3
      if vostatus eq 1 then
        set var voplname to .vowlname
        set var vopfname to .vowfname
        set var #return to pgdn
      else
        set var voplname to unk
        set var vopfname to unk
        write "Press [PGDN] after entering initial information to +
check if" at 1 5
        write "operator exists in database.  Press [ESC] to quit" +
        at 2 10
        draw cg4100a5 with all at 4
        edit var return esc pgdn
      endif
    endif
    if #return ne esc then
      if review eq N and backup eq N then
        while voplname exists and vopfname exists then
          set pointer #3 e3 for operator where oplname = .voplname and +
          opfname = .vopfname
          newpage
          while e3 eq 0 then
            *(operator last/first name exist in database)
            set var vopmi to opmi in #3
            set var vopdob to opdob in #3
            set var vopaddr to opaddr in #3
            set var vopcity to opcity in #3
            set var vopst to opstate in #3
            set var vopzip to opzip in #3
            set var voptelno to optelno in #3
            set var vopcrse to opcourse in #3
            newpage
            draw cg4100a5 with all at 3
            draw cg4100a6 with all at 8
            write "Correct operator? (N)" at 21 20
            fillin opfound using "(" at 21 38
            if opfound exists and opfound eq Y then
              set var op1 to .voplname
              set var op2 to .vopfname
              set var op3 to opmi in #3
              set var op4 to opaddr in #3
              set var op5 to opcity in #3
              set var op6 to opstate in #3
              set var op7 to opzip in #3
              set var op8 to optelno in #3
              set var op9 to opdob in #3
              set var op10 to opcourse in #3
              set var op11 to opid in #3
              set var vopid to opid in #3
              break
            endif
            next #3 e3
          endwhile
          if e3 ne 0 then
            newpage
            write "This operator does not exist in the database" at 5 5
            fillin go using "Press [ENTER] to continue" at 9 5
            if vostatus eq 1 then
              set var vopmi to .vowmi
              set var vopaddr to .vowaddr
              set var vopcity to .vowcity
              set var vopst to .vowst
              set var vopzip text
              set var vopzip to .vowzip
              set var voptelno text
              set var voptelno to .vowtelno
              set var vopcrse text
              set var vopdob to .vowdob
            else
              clear vopmi
              clear vopaddr
              clear vopcity
              set var vopst to CA
              clear vopzip
              clear voptelno
              set var vopdob to 01/01/1901
            endif
            clear vopcrse
            compute vmaxop as max opid from operator
            if vmaxop fails then
              set var vmaxop to 0
            endif
            set var vopid to .vmaxop + 1
          endif
          break
        endwhile
      endif
      newpage
      write "Press [ESC] to quit, [PGUP] to move up, [PGDN] to move +
down a page" at 1 1
      draw cg4100a5 with all at 3
      draw cg4100a6 with all at 8
      if vostatus eq 1 then
        edit var vopcrse return esc pgup pgdn
      else
        edit var return esc pgup pgdn
      endif
    endif
  endif
  if pg eq 4 then
    *(enter boarding location information)
    if review eq N and backup eq N then
      set var pgcount to 4
      set var vobsbody to text
      set var vobsbody to "Monterey Bay"
      set var vobcount to Monterey
      set var vobscity to Monterey
      set var vobstate to CA
      set var vobslat to 00.000
      set var vobslong to 000.000
    endif
    write "Press [ESC] to quit, [PGUP] to move up, [PGDN] to move down +
a page" at 1 1
    draw cg4100a7 with all at 3
    edit var return esc pgup pgdn
  endif
  if pg eq 5 then
    *(enter boarding violation/unsafe condition information)
    if review eq N and backup eq N then
      set var pgcount to 5
    endif
    write "Press [ESC] to quit, [PGUP] to move up, [PGDN] to move +
down a page" at 1 1
    draw cg4100a8 with all at 3
    edit var return esc pgup pgdn
  endif
  if pg eq 6 then
    *(enter boarding officer information)
    if review eq N and backup eq N then
      set var pgcount to 6
      set var vbounit text
      set var vbounit to "CG Group Monterey"
      set var vbodist text
      set var vbodist to 11
      set var vboopfac text
      set var vboopfac to 36268
      set var vbosrise to 06:00:00
      set var vbosset to 19:00:00
      set var vweapons to 0
    endif
    write "Press [ESC] to quit, [PGUP] to move up, [PGDN] to move +
down a page" at 1 1
    draw cg4100a9 with all at 3
    edit var return esc pgup pgdn
  endif
  if pg eq 7 then
    *(enter additional vessel information)
    if review eq N and backup eq N then
      set var pgcount to 7
      if el ne 0 then
        set var vvalue to $0
        set var vmasts text
        set var vmasts to 0
        set var vslcolor to unk
        set var vhlcolor to unk
        set var vtrcolor to unk
        set var vcbcolor to unk
        set var vdkcolor to unk
      endif
    endif
    write "Press [ESC] to quit, [PGUP] to move up, [PGDN] to move +
down a page" at 1 1
    draw sbdform1 with all at 3
    edit var voffno vhome vcall vvalue vsuper vmasts vslcolor vsailno +
    vhlcolor vtrcolor vcbcolor vdkcolor vmarks return esc pgup pgdn
  endif
  if pg eq 8 then
    *(enter general sar information)
    if review eq N and backup eq N then
     set var pgcount to 8
     set var vsardate to .vbodate
     set var vsartime to .vbotime
     set var vsarlat to .vobslat
     set var vsarlong to .vobslong
     set var vdistoff to 0
     set var vsarpob to .vvslpob
     set var vsarday to .vboday
    endif
    write "Press [ESC] to quit, [PGUP] to move up, [PGDN] to move +
down a page" at 1 1
    draw sarform1 with all at 3
    edit var vfoldno vucn vmucn vsardate vsarday vsartime vsarnod +
    vimmed vsarlat vsarlong vdistoff vsarpob return esc pgdn
  endif
  if pg eq 9 then
    *(enter sar weather information)
    if review eq N and backup eq N then
      set var pgcount to 9
      set var vsunrise to .vbosrise
      set var vsunset to .vbosset
      set var vceiling to 0
      set var vclouds to 0
      set var vvisdist to 0
      set var vwinddir to 0
      set var vwindspd to 0
      set var vseas to 0
      set var vswells to 0
    endif
    write "Press [ESC] to quit or [PGUP] to move up a page" at 1 1
    draw sarform9 with all at 3
    edit var vsunrise vsunset vceiling vclouds vvisdist vwinddir +
    vwindspd vseas vswells return esc pgup
    set var loadmenu to "Y"
  endif
  if #return eq pgup then
    set var backup to Y
    if pg gt 1 then
      set var pg to .pg - 1
    else
      set var pg to 1
    endif
  endif
  if #return eq pgdn then
```

```
    if pg lt 9 then
      set var pg to .pg + 1
      if pgcount lt .pg then
       set var backup to N
      endif
    else
      set var pg to 9
    endif
  endif
  if #return eq esc then
    set var pg to 0
    set var backup to N
  endif
endwhile  *(end of main loop)
newpage
write "Program loading in progress.  Please stand by ......." at 10 10
if loadmenu eq "Y" then
  set var choice to 0
  while choice lt 1 or choice gt 3 then
    newpage
    write "Data Load Options" at 2 20
    write "-----------------" at 3 20
    write "Enter:" at 5 20
    write "1-To load the data into the database" at 6 20
    write "2-To ignore the data entered" at 7 20
    write "3-To go back and verify data entered" at 8 20
    write "Your choice: " at 10 20
    fillin choice using " " at 10 33
    newpage
    if choice eq 1 then
      *(test for required information)
      set var test to "F"
      if vbodate exists and vbotime exists and vbono exists and +
      vvslname exists and vobslat exists and vobslong exists then
        if vsardate exists and vfoldno exists and vucn exists and +
        vvslname exists and vsarlat exists and vsarlong exists and +
        vsarnod exists and vimmed exists then
          if vviol54 exists or vviol55 exists or vviol56 exists or +
          vviol57 exists or vviol58 exists or vviol59 exists or +
          vviol60 exists or vviol61 exists or vviol62 exists or +
          vviol63 exists then
            set var test to "T"
          endif
          if vviol64 exists or vviol65 exists or vvicl66 exists or +
          vviol67 exists or vviol68 exists or vuns69 exists or vuns70 +
          exists or vuns71 exists or vuns72 exists or vuns73 exists then
            set var test to "T"
          endif
          if vuns74 exists or vuns75 exists or vuns76 exists or vuns77 +
          exists or vuns78 exists or vuns79 exists then
            set var test to "T"
          endif
          if test eq "T" then
            set error messages off
            if el eq 0 then
              *(vessel exists in database)
              write "changing values in vessel"
              delete rows from #1
            else
              write "loading vessel"
            endif
            load vessel
              .vvslno .vvslname .vvslhin .vvslmake .vvslmod .vvslyear +
              .vvslton .vvslft .vvslin .vvslhp .vvsluse .vvsltype +
              .vvslprop .vvslhull .vvsleng .vvslfuel .vconstr .vdecal +
              .vcmeyear .voffno .vhome .vcall .vvalue .vsuper .vmasts +
              .vsailno .vhlcolor .vtrcolor .vslcolor .vcbcolor +
              .vdkcolor .vmarks
            end
            if el eq 0 then
  set pointer #1 el for vessel whe vslhin eq .v3
  if el ne 0 then
    load vessel
      .v1 .v2 .v3 .v4 .v5 .v6 .v7 .v8 .v9 .v10 .v11 .v12 +
      .v13 .v14 .v15 .v16 .v17 .v18 .v19 .v20 .v21 .v22 +
      .v23 .v24 .v25 .v26 .v27 .v28 .v29 .v30 .v31 .v32
    end
  endif
endif
write "loading owvsl"
load owvsl
  .vvslhin .vownid
end
write "loading opvsl"
load opvsl
  .vvslhin .vopid
end
if vostatus eq 1 then
  load opown
    .vownid .vopid
  end
endif
set pointer #1 el for boarding where bono = .vbono
if el eq 0 then
  set var difbo to "F"
  set var b1 to bodate in #1
  set var b2 to botime in #1
  set var b3 to obslat in #1
  set var b4 to obslong in #1
  if b1 ne .vbodate or b2 ne .vbotime or b3 ne .vobslat or +
  b4 ne .vobslong then
    set var difbo to "T"
  endif
else
  set var difbo to "T"
endif
if difbo eq T then
  write "loading boarding"
  load boarding
    .vbodate .vbotime .vbono .vvslpob .vadult .vchild +
    .vostatus .vobsbody .vobcount .vobscity .vobstate +
    .vobslat .vobslong .vviol54 .vviol55 .vviol56 +
    .vviol57 .vviol58 .vviol59 .vviol60 .vviol61 .vviol62 +
    .vviol63 .vviol64 .vviol65 .vviol66 .vviol67 .vviol68 +
    .vuns69 .vuns70 .vuns71 .vuns72 .vuns73 .vuns74 +
    .vuns75 .vuns76 .vuns77 .vuns78 .vuns79 .vborem1 +
    .vborem2 .vborem3 .vboname .vborate .vbounit .vbodist +
    .vboopfac .vboday .vbosrise .vbosset .vvslhin +
    .vweapons
  end
endif
set pointer #1 el for sar where folderno = .vfoldno
if el eq 0 then
  set var difsar to "F"
  set var s1 to sardate in #1
  set var s2 to sartime in #1
  set var s3 to sarlat in #1
  set var s4 to sarlong in #1
  set var s5 to sarnod in #1
  set var s6 to sarvhin in #1
  if s1 ne .vsardate or s2 ne .vsartime or s3 ne .vsarlat +
  or s4 ne .vsarlong or s5 ne .vsarnod or s6 ne .vvslhin +
  then
    set var difsar to "T"
  endif
else
  set var difsar to "T"
endif
if difsar eq T then
  write "loading sar"
  load sar
    .vfoldno .vucn .vmucn .vsardate .vsarday .vsartime +
    .vsunrise .vsunset .vsarnod .vsarpob .vvslhin +
    .vsarlat .vsarlong .vdistoff .vimmed .vwinddir +
    .vwindspd .vvisdist .vclouds .vceiling .vseas .vswells
  end
endif
if e2 eq 0 then
  *(owner exists in database)
  write "changing values in owner"
  delete rows from #2
else
  write "loading owner"
endif
load owners
  .vowlname .vowfname .vowmi .vowaddr .vowcity .vowst +
  .vowzip .vowtelno .vowdob .vownid
end
if e2 eq 0 then
  set pointer #2 e2 for owners whe ownid eq .ow10
  if e2 ne 0 then
    load owners
      .ow1 .ow2 .ow3 .ow4 .ow5 .ow6 .ow7 .ow8 .ow9 .ow10
    end
  endif
endif
if e3 eq 0 then
  *(operator exists in database)
  write "changing values in operator"
  delete rows from #3
else
  write "loading operator"
endif
load operator
  .voplname .vopfname .vopmi .vopaddr .vopcity .vopst +
  .vopzip .voptelno .vopdob .vopcrse .vopid
end
if e3 eq 0 then
  set pointer #3 e3 for operator where opid eq .op11
  if e3 ne 0 then
    load operator
      .op1 .op2 .op2 .op3 .op4 .op5 .op6 .op7 .op8 .op9 +
      .op10 .op11
    end
  endif
endif
write "  "
set error messages on
*(return option in case of entry/edit errors)
write "Return to correct data entered? (Y)" at 20 10
write "Note:  correct only errors indicated by messages" +
at 22 5
fillin er using "(" at 20 42
if er fails or er eq "Y" then
  set var review to "Y"
  set var pg to 1
  set pointer #1 e1 for vessel where vslhin eq .vvslhin
  set pointer #2 e2 for owners where ownid eq .vownid
  set pointer #3 e3 for operator where opid eq .vopid
  break
else
  newpage
  write "Removing duplicate entries from tables" at 10 10
  write "Please stand by................" at 15 12
  delete duplicates from vessel
  delete duplicates from owvsl
  delete duplicates from opvsl
  delete duplicates from opown
  delete duplicates from owners
  delete duplicates from operator
```

```
        endif
      endif
    endif
  else
    newpage
    write "Load not performed.  Required information missing." +
    at 5 20
    write "Required information includes:" at 10 30
    write "Boarding Date, Boarding Time, Boarding Number" at 12 35
    write "Vessel Name, Boarding Latitude, Boarding Longitude" +
    at 13 35
    write "A Violation or Unsafe Condition" at 14 35
    write "Sar Date, Folder Number, Unit Case Number" at 15 35
    write "Vessel Name, Sar Latitude, Sar Longitude" at 16 35
    write "Sar Nature of Distress" at 17 35
    fillin go using "Press [ENTER] to return to data entry" +
    at 18 25
    set var pg to 1
    set var review to "Y"
    break
  endif
endif
if choice eq 1 or choice eq 2 then
  *(enter sar units)
  newpage
  write "Enter sar units? (Y)" at 12 20
  fillin s_units using "(" at 12 37
  if s_units fails or s_units eq Y then
    set var more to "Y"
    while more eq Y then
      set var vuwtime to .vsartime
      set var vuwdate to .vsardate
      set var vetadate to .vsardate
      set var vdateos to .vsardate
      set var valgdate to .vsardate
      set var vtowdate to .vsardate
      set var vdvmoord to .vsardate
      set var vsumoord to .vsardate
      newpage
      write "Press [PGDN] to load data or [ESC] to quit" at 1 1
      draw sarform8 with all at 3
      edit var vunit vuwtime vuwdate voseta vetadate vostime +
      vdateos valgtime valgdate vtowtime vtowdate vdvmoort +
      vdvmoord vsumoort vsumoord return esc pgdn
      if #return eq pgdn then
        if voseta fails then
          clear vetadate
          set var vetadate date
        endif
        if vostime fails then
          clear vdateos
          set var vdateos date
        endif
        if valgtime fails then
          clear valgdate
          set var valgdate date
        endif
        if vtowtime fails then
          clear vtowdate
          set var vtowdate date
        endif
        if vdvmoort fails then
          clear vdvmoord
          set var vdvmoord date
        endif
        if vsumoort fails then
          clear vsumoord
          set var vsumoord date
        endif
        set pointer #1 e1 for sarunits whe foldno eq .vfoldno and +
            unitname eq .vunit
            newpage
            if e1 eq 0 then
              write "Duplicate entry not loaded"
            else
              write "Loading sarunits"
              load sarunits
                .vfoldno .vunit .vuwtime .vuwdate .voseta .vetadate +
                .vostime .vdateos .valgtime .valgdate .vtowtime +
                .vtowdate .vdvmoort .vdvmoord .vsumoort .vsumoord
              end
            endif
            write "More units to enter? (N)" at 23 20
            fillin mrunits using "(" at 23 41
            if mrunits fails or mrunits eq N then
              set var more to N
            else
              clear vunit
              clear voseta
              clear vostime
              clear valgtime
              clear vtowtime
              clear vdvmoort
              clear vsumoort
            endif
          endif
          if #return eq esc then
            set var more to N
          endif
        endwhile
      endif
      *(enter additional owners)
      newpage
      write "Enter additional owners? (N)" at 12 20
      fillin addl_own using "(" at 12 45
      if addl_own exists and addl_own eq Y then
        set var more to "Y"
        while more eq Y then
          set var vowlname to unk
          set var vowfname to unk
          newpage
          write "Press [PGDN] after entering initial information to +
check if" at 1 5
          write "owner exists in database.  Press [ESC] to quit" at 2 10
          draw cg4100a3 with all at 4
          edit var return esc pgdn
          if #return ne esc then
            while vowlname exists and vowfname exists then
              set pointer #2 e2 for owners where ownlname = .vowlname +
              and ownfname = .vowfname
              newpage
              while e2 eq 0 then
                *(owner last/first name exist in database)
                set var vowmi to ownmi in #2
                set var vowdob to owndob in #2
                set var vowaddr to ownaddr in #2
                set var vowcity to owncity in #2
                set var vowst to ownstate in #2
                set var vowzip to ownzip in #2
                set var vowtelno to owntelno in #2
                newpage
                draw cg4100a3 with all at 3
                draw cg4100a4 with all at 8
                write "Correct owner? (N)" at 21 20
                fillin owfound using "(" at 21 35
                if owfound exists and owfound eq Y then
                  set var ow1 to .vowlname
                  set var ow2 to .vowfname
                  set var ow3 to ownmi in #2
                  set var ow4 to ownaddr in #2
      set var ow5 to owncity in #2
      set var ow6 to ownstate in #2
      set var ow7 to ownzip in #2
      set var ow8 to owntelno in #2
      set var ow9 to owndob in #2
      set var ow10 to ownid in #2
      set var vownid to ownid in #2
      break
    endif
    next #2 e2
  endwhile
  if e2 ne 0 then
    newpage
    writ( "This owner does not exist in the database" +
    at 5 5
    fillin go using "Press [ENTER] to continue" at 9 5
    clear vowmi
    clear vowaddr
    clear vowcity
    set var vowst to CA
    clear vowzip
    clear vowtelno
    set var vowdob to 01/01/1901
    clear vostatus
    compute vmaxown as max ownid from owners
    if vmaxown fails then
      set var vmaxown to 0
    endif
    set var vownid to .vmaxown + 1
  endif
  break
endwhile
label redoown
newpage
write "Press [PGDN] to load data or [ESC] to quit" at 1 1
draw cg4100a3 with all at 3
draw sarform5 with all at 8
edit var return esc pgup pgdn
if #return eq pgdn then
  set error messages off
  newpage
  if e2 eq 0 then
    *(owner exists in database)
    write "changing values in owner"
    delete rows from #2
  else
    write "loading owner"
  endif
  load owners
    .vowlname .vowfname .vowmi .vowaddr .vowcity .vowst +
    .vowzip .vowtelno .vowdob .vownid
  end
  if e2 eq 0 then
    set pointer #2 e2 for owners whe ownid eq .ow10
    if e2 ne 0 then
      load owners
        .ow1 .ow2 .ow3 .ow4 .ow5 .ow6 .ow7 .ow8 .ow9 .ow10
      end
    endif
  endif
  set error messages on
  write "Return to correct errors? (N)" at 15 10
  write "Note:  correct only errors indicated by messages" +
  at 20 5
  fillin fixown using "(" at 15 36
  if fixown exists and fixown eq Y then
    goto redoown
  endif
  set pointer #2 e2 for owvsl where vhinown eq .vvslhin +
  and idowner eq .vownid
```

```
              *(check if record already exists)
              newpage
              if e2 ne 0 then
                load owvsl
                  .vvslhin .vownid
                end
              endif
            endif
          endif
          write "More owners to enter? (N)" at 20 20
          fillin mrownrs using "(" at 20 42
          if mrownrs fails or mrownrs eq N then
            set var more to N
          endif
        endwhile
      endif
      *(enter additional operators)
      newpage
      write "Enter additional operators? (N)" at 12 20
      fillin addl_opr using "(" at 12 48
      if addl_opr exists and addl_opr eq Y then
        set var more to "Y"
        while more eq Y then
          set var voplname to unk
          set var vopfname to unk
          newpage
          write "Press [PGDN] after entering initial information to
check if" at 1 5
          write "operator exists in database.  Press [ESC] to quit" +
          at 2 10
          draw cg4100a5 with all at 4
          edit var return esc pgdn
          if #return ne esc then
            while voplname exists and vopfname exists then
              set pointer #3 e3 for operator where oplname = .voplname +
              and opfname = .vopfname
              newpage
              while e3 eq 0 then
                *(operator last/first name exist in database)
                set var vopmi to opmi in #3
                set var vopdob to opdob in #3
                set var vopaddr to opaddr in #3
                set var vopcity to opcity in #3
                set var vopst to opstate in #3
                set var vopzip to opzip in #3
                set var voptelno to optelno in #3
                set var vopcrse to opcourse in #3
                newpage
                draw cg4100a5 with all at 3
                draw cg4100a6 with all at 8
                write "Correct operator? (N)" at 21 20
                fillin opfound using "(" at 21 39
                if opfound exists and opfound eq Y then
                  set var op1 to .voplname
                  set var op2 to .vopfname
                  set var op3 to opmi in #3
                  set var op4 to opaddr in #3
                  set var op5 to opcity in #3
                  set var op6 to opstate in #3
                  set var op7 to opzip in #3
                  set var op8 to optelno in #3
                  set var op9 to opdob in #3
                  set var op10 to opcourse in #3
                  set var op11 to opid in #3
                  set var vopid to opid in #3
                  break
                endif
                next #3 e3
              endwhile
              if e3 ne 0 then
      newpage
      write "This operator does not exist in the database"
      at 5 5
      fillin go using "Press [ENTER] to continue" at 9 5
      clear vopmi
      clear vopaddr
      clear vopcity
      set var vopst to CA
      clear vopzip
      clear voptelno
      clear vopcrse
      set var vopdob to 01/01/1901
      compute vmaxoper as max opid from operator
      if vmaxoper fails then
        set var vmaxoper to 0
      endif
      set var vopid to .vmaxoper + 1
    endif
    break
  endwhile
  label redooper
  newpage
  write "Press [PGDN] to load data or [ESC] to quit" at 1 1
  draw cg4100a5 with all at 3
  draw cg4100a6 with all at 8
  edit var return esc pgup pgdn
  if #return eq pgdn then
    newpage
    set error messages off
    if e3 eq 0 then
      *(operator exists in database)
      write "changing values in operator"
      delete rows from #3
    else
      write "loading operator"
    endif
    load operator
      .voplname .vopfname .vopmi .vopaddr .vopcity .vopst +
      .vopzip .voptelno .vopdob .vopcrse .vopid
    end
    if e3 eq 0 then
      set pointer #3 e3 for operator where opid eq .op11
      if e3 ne 0 then
        load operator
          .op1 .op2 .op2 .op3 .op4 .op5 .op6 .op7 .op8 .op9 +
          .op10 .op11
        end
      endif
    endif
    set error messages on
    write "Return to correct errors? (N)" at 15 10
    write "Note:  correct only errors indicated by messages" +
    at 20 5
    fillin fixopr using "(" at 15 36
    if fixopr exists and fixopr eq Y then
      goto redooper
    endif
    set pointer #3 e3 for opvsl where vhinoper eq .vvslhin +
    and idoper eq .vopid
    *(check if record already exists)
    newpage
    if e3 ne 0 then
      load opvsl
        .vvslhin .vopid
      end
    endif
  endif
endif
write "More operators to enter? (N)" at 20 20
fillin mropers using "(" at 20 45
          if mropers fails or mropers eq N then
            set var more to N
          endif
        endwhile
      endif
      newpage
      write "Enter another sar with boarding? (Y)" at 12 20
      fillin cont using "(" at 12 53
      if cont fails or cont eq Y then
        *(reset variables to start condition)
        clear all variables
        set var pg to 1
        set var review to "N"
        set var loadmenu to "N"
        break
      else
        set null -0-
        set var pg to 0
      endif
    endif
    if choice eq 3 then
      set var review to "Y"
      set var pg to 1
    endif
  endwhile
else
  newpage
  write "Enter another sar with boarding? (Y)" at 12 20
  fillin cont using "(" at 12 53
  if cont fails or cont eq Y then
    *(reset variables to start condition)
    clear all variables
    set var pg to 1
    set var review to "N"
    set var loadmenu to "N"
  else
    set error messages off
    set null -0-
    set var pg to 0
  endif
endif
if pg eq 1 then
  goto begin
endif
return
*( end )


$COMMAND
ADD_SUT
*(*********************************************************************

    Program:           Add Sar Units
    Author:            H R Lang and J D Allen
    Date Written:      August 1987
    Description:       This program allows the user to input sar
                       unit data.
    Tables Used:       sar, sarunits
    Temp Tables:       none
    Reports Used:      none
    Forms Used:        sarform8

**********************************************************************)

set error messages on
set messages off
set escape off
newpage
```

```
write "Program loading in progress.  Please stand by ........" at 10 10
clear all variables
set null ?
set var more to Y
set var vfoldno text
set var vunit text
while more eq Y then  *(main loop)
  set var done to Y
  newpage
  write "Enter folder number of sar case: " at 10 5
  fillin vfoldno using " " at 10 37
  write "Enter name of sar unit to be added: " at 15 5
  fillin vunit using " " at 15 40
  if vfoldno exists and vunit exists then
    set pointer #1 e1 for sar where folderno eq .vfoldno
    if e1 eq 0 then
      set pointer #2 e2 for sarunits where foldno eq .vfoldno and +
      unitname eq .vunit
      if e2 eq 0 then
        newpage
        write "This record already exists in the database" at 15 15
        write "Press any key to continue" at 18 20
        pause
      else
        set var vuwtime to sartime in #1
        set var vuwdate to sardate in #1
        set var vetadate to sardate in #1
        set var vdateos to sardate in #1
        set var valgdate to sardate in #1
        set var vtowdate to sardate in #1
        set var vdvmoord to sardate in #1
        set var vsumoord to sardate in #1
        newpage
        write "Press [PGDN] to load data or [ESC] to quit" at 1 1
        draw sarform8 with all at 3
        edit var vunit vuwtime vuwdate voseta vetadate vostime vdateos +
        valgtime valgdate vtowtime vtowdate vdvmoort vdvmoord vsumoort +
        vsumoord return esc pgdn
        if #return eq pgdn then
          if voseta fails then
            clear vetadate
            set var vetadate date
          endif
          if vostime fails then
            clear vdateos
            set var vdateos date
          endif
          if valgtime fails then
            clear valgdate
            set var valgdate date
          endif
          if vtowtime fails then
            clear vtowdate
            set var vtowdate date
          endif
          if vdvmoort fails then
            clear vdvmoord
            set var vdvmoord date
          endif
          if vsumoort fails then
            clear vsumoord
            set var vsumoord date
          endif
          load sarunits
            .vfoldno .vunit .vuwtime .vuwdate .voseta .vetadate +
            .vostime .vdateos .valgtime .valgdate .vtowtime +
            .vtowdate .vdvmoort .vdvmoord .vsumoort .vsumoord
          end
        endif
      endif
    else
      newpage
      write "This folder number does not exist in the database" at 10 10
      write "Do you wish to reenter folder number? (Y)" at 15 10
      fillin reenter using "(" at 15 48
      if reenter fails or reenter eq Y then
        set var done to N
      endif
    endif
  endif
  if done eq Y then
    newpage
    write "More sar units to enter? (N)" at 15 20
    fillin mrunits using "(" at 15 45
    if mrunits fails or mrunits eq N then
      set var more to N
    else
      clear all variables
      set var more to Y
    endif
  endif
endwhile  *(end main loop)
set error messages off
set null -0-
return
*( end )


$COMMAND
ADD_BRD
*(**********************************************************************

    Program:               Add Boarding
    Author:                H R Lang and J D Allen
    Date Written:          July 1987
    Description:           This program allows the user to input
                           boarding information.
    Tables Used:           boarding, vessel, owners, operator, owvsl,
                           opvsl
    Temp Tables:           none
    Reports Used:          none
    Forms Used:            cg4100a1, cg4100a2, cg4100a3, cg4100a4,
                           cg4100a5, cg4100a6, cg4100a7, cg4100a8,
                           cg4100a9, sarform5

**********************************************************************)

set error messages on
set messages off
set escape off
newpage
write "Program loading in progress.  Please stand by ........" at 10 10
clear all variables
set null ?
set var pg to 1
*(initialize edit and load control variables)
set var review to N
set var loadmenu to N
label begin
set var backup to N
set var fixhin to N
set var pgcount to 0
while pg gt 0 then  *(main loop)
  newpage
  if pg eq 1 then
    *(enter boarding/vessel information)
    if review eq N and backup eq N then
      set var pgcount to 1
  set var vbodate to .#date
  set var vbotime to 12:00:00
  set var vvslname to unk
  write "Press [PGDN] after entering initial information to check +
  if" at 1 5
  write "vessel exists in database.  Press [ESC] to quit" at 2 10
  draw cg4100a1 with all at 4
  edit var return esc pgdn
else
  write "Press [PGDN] after review/correction or [ESC] to quit" +
  at 1 5
  draw cg4100a1 with all at 4
  if fixhin eq N then
    edit var vvslname vbodate vbotime vbono vvslno return esc pgdn
  else
    edit var return esc pgdn
  endif
endif
if #return ne esc then
  if review eq N and backup eq N or fixhin eq Y then
    set var pgcount to 1
    while vvslhin exists then
      set pointer #1 e1 for vessel where vslhin = .vvslhin
      newpage
      set var vvslpob to 0
      set var vadult to 0
      set var vchild to 0
      if e1 eq 0 then
        *(vessel exists in database)
        set var vvslno to vslno in #1
        set var vvslname to vslname in #1
        set var vvslmake to vslmake in #1
        set var vvslmod to vslmodel in #1
        set var vvslyear to vslyear in #1
        set var vvslton to vsltons in #1
        set var vvslft to vsllenft in #1
        set var vvslin to vsllenin in #1
        set var vvslhp to vslhp in #1
        set var vvsluse to vsluse in #1
        set var vvsltype to vsltype in #1
        set var vvslprop to vslprop in #1
        set var vvslhull to vslhull in #1
        set var vvsleng to vsleng in #1
        set var vvslfuel to vslfuel in #1
        set var vconstr to vslconst in #1
        set var vdecal to cmedecal in #1
        set var vcmeyear to cmeyear in #1
        set var voffno to vsloffno in #1
        set var vhome to homeport in #1
        set var vcall to vslcall in #1
        set var vvalue to vslvalue in #1
        set var vsuper to vslsuper in #1
        set var vmasts to vslmasts in #1
        set var vsailno to sailno in #1
        set var vhlcolor to hlcolor in #1
        set var vtrcolor to trcolor in #1
        set var vslcolor to slcolor in #1
        set var vcbcolor to cbcolor in #1
        set var vdkcolor to dkcolor in #1
        set var vmarks to vslmarks in #1
        set var v1 to vslno in #1
        set var v2 to vslname in #1
        set var v3 to vslhin in #1
        set var v4 to vslmake in #1
        set var v5 to vslmodel in #1
        set var v6 to vslyear in #1
        set var v7 to vsltons in #1
        set var v8 to vsllenft in #1
        set var v9 to vsllenin in #1
        set var v10 to vslhp in #1
```

```
            set var v11 to vsluse in #1
            set var v12 to vsltype in #1
            set var v13 to vslprop in #1
            set var v14 to vslhull in #1
            set var v15 to vsleng in #1
            set var v16 to vslfuel in #1
            set var v17 to vslconst in #1
            set var v18 to cmedecal in #1
            set var v19 to cmeyear in #1
            set var v20 to vsloffno in #1
            set var v21 to homeport in #1
            set var v22 to vslcall in #1
            set var v23 to vslvalue in #1
            set var v24 to vslsuper in #1
            set var v25 to vslmasts in #1
            set var v26 to sailno in #1
            set var v27 to hlcolor in #1
            set var v28 to trcolor in #1
            set var v29 to slcolor in #1
            set var v30 to cbcolor in #1
            set var v31 to dkcolor in #1
            set var v32 to vslmarks in #1
          else
            *(vessel does not exist in database)
            set var vvslmake to unk
            set var vvslmod to unk
            set var vvslyear text
            set var vvslyear to 1901
            set var vvslhp text
            set var vvslhp to 0
            set var vvsluse text
            set var vvsluse to 0
            set var vvsltype text
            set var vvsltype to 9
            set var vvslprop text
            set var vvslprop to 9
            set var vvslhull text
            set var vvslhull to 8
            set var vvsleng text
            set var vvsleng to 0
            set var vvslfuel text
            set var vvslfuel to 0
            set var vconstr text
            set var vconstr to 0
            set var vvslton to 0
            set var vvslft to 0
            set var vvslin to 0
            set var vdecal to N
          endif
          break
        endwhile
        newpage
        write "Press [ESC] to quit or [PGDN] to move down a page" at 1 1
        draw cg4100a1 with all at 3
        draw cg4100a2 with all at 10
        edit var return esc pgdn
      endif
    endif
  endif
  if pg eq 2 then
   *(enter owner information)
    if review eq N and backup eq N then
      set var pgcount to 2
      set var vowlname to unk
      set var vowfname to unk
      write "Press [PGDN] after entering initial information to check +
if" at 1 5
      write "owner exists in database.  Press [ESC] to quit" at 2 10
      draw cg4100a3 with all at 4
      edit var return esc pgdn
    endif
    if #return ne esc then
      if review eq N and backup eq N then
        while vowlname exists and vowfname exists then
          set pointer #2 e2 for owners where ownlname = .vowlname and +
          ownfname = .vowfname
          newpage
          while e2 eq 0 then
            *(owner last/first name exist in database)
            set var vowmi to ownmi in #2
            set var vowdob to owndob in #2
            set var vowaddr to ownaddr in #2
            set var vowcity to owncity in #2
            set var vowst to ownstate in #2
            set var vowzip to ownzip in #2
            set var vowtelno to owntelno in #2
            newpage
            draw cg4100a3 with all at 3
            draw cg4100a4 with all at 8
            write "Correct owner? (N)" at 21 20
            fillin owfound using "(" at 21 35
            if owfound exists and owfound eq Y then
              set var ow1 to .vowlname
              set var ow2 to .vowfname
              set var ow3 to ownmi in #2
              set var ow4 to ownaddr in #2
              set var ow5 to owncity in #2
              set var ow6 to ownstate in #2
              set var ow7 to ownzip in #2
              set var ow8 to owntelno in #2
              set var ow9 to owndob in #2
              set var ow10 to ownid in #2
              set var vownid to ownid in #2
              break
            endif
            next #2 e2
          endwhile
          if e2 ne 0 then
            newpage
            write "This owner does not exist in the database" at 5 5
            fillin go using "Press [ENTER] to continue" at 9 5
            clear vowmi
            clear vowaddr
            clear vowcity
            set var vowst to CA
            clear vowzip
            clear vowtelno
            set var vowdob to 01/01/1901
            clear vostatus
            compute vmaxown as max ownid from owners
            if vmaxown fails then
              set var vmaxown to 0
            endif
            set var vownid to .vmaxown + 1
          endif
          break
        endwhile
      endif
      newpage
      write "Press [ESC] to quit, [PGUP] to move up, [PGDN] to move +
down a page" at 1 1
      draw cg4100a3 with all at 3
      draw cg4100a4 with all at 8
      edit var return esc pgup pgdn
    endif
  endif
  if pg eq 3 then
   *(enter operator information)
    if review eq N and backup eq N then
      set var pgcount to 3
      if vostatus eq 1 then
        set var voplname to .vowlname
        set var vopfname to .vowfname
        set var #return to pgdn
      else
        write "Press [PGDN] after entering initial information to +
check if" at 1 5
        write "operator exists in database.  Press [ESC] to quit" +
        at 2 10
        set var voplname to unk
        set var vopfname to unk
        draw cg4100a5 with all at 4
        edit var return esc pgdn
      endif
    endif
    if #return ne esc then
      if review eq N and backup eq N then
        while voplname exists and vopfname exists then
          set pointer #3 e3 for operator where oplname = .voplname and +
          opfname = .vopfname
          newpage
          while e3 eq 0 then
            *(operator last/first name exist in database)
            set var vopmi to opmi in #3
            set var vopdob to opdob in #3
            set var vopaddr to opaddr in #3
            set var vopcity to opcity in #3
            set var vopst to opstate in #3
            set var vopzip to opzip in #3
            set var voptelno to optelno in #3
            set var vopcrse to opcourse in #3
            newpage
            draw cg4100a5 with all at 3
            draw cg4100a6 with all at 8
            write "Correct operator? (N)" at 21 20
            fillin opfound using "(" at 21 38
            if opfound exists and opfound eq Y then
              set var op1 to .voplname
              set var op2 to .vopfname
              set var op3 to opmi in #3
              set var op4 to opaddr in #3
              set var op5 to opcity in #3
              set var op6 to opstate in #3
              set var op7 to opzip in #3
              set var op8 to optelno in #3
              set var op9 to opdob in #3
              set var op10 to opcourse in #3
              set var op11 to opid in #3
              set var vopid to opid in #3
              break
            endif
            next #3 e3
          endwhile
          if e3 ne 0 then
            newpage
            write "This operator does not exist in the database" at 5 5
            fillin go using "Press [ENTER] to continue" at 9 5
            if vostatus eq 1 then
              set var vopmi to .vowmi
              set var vopaddr to .vowaddr
              set var vopcity to .vowcity
              set var vopst to .vowst
              set var vopzip to text
              set var vopzip to .vowzip
              set var voptelno to text
              set var voptelno to .vowtelno
              set var vopdob to .vowdob
            else
              clear vopmi
              clear vopaddr
              clear vopcity
              set var vopst to CA
              clear vopzip
              clear voptelno
              set var vopdob to 01/01/1901
            endif
            clear vopcrse
            compute vmaxop as max opid from operator
            if vmaxop fails then
              set var vmaxop to 0
            endif
            set var vopid to .vmaxop + 1
          endif
          break
        endwhile
      endif
      newpage
      write "Press [ESC] to quit, [PGUP] to move up, [PGDN] to move +
down a page" at 1 1
      draw cg4100a5 with all at 3
      draw cg4100a6 with all at 8
      if vostatus eq 1 then
        edit var vopcrse return esc pgup pgdn
      else
        edit var return esc pgup pgdn
      endif
    endif
  endif
  if pg eq 4 then
    *(enter boarding location information)
    if review eq N and backup eq N then
      set var pgcount to 4
      set var vobsbody to text
      set var vobsbody to "Monterey Bay"
      set var vobcount to Monterey
      set var vobscity to Monterey
      set var vobstate to CA
      set var vobslat to 00.000
      set var vobslong to 000.000
    endif
    write "Press [ESC] to quit, [PGUP] to move up, [PGDN] to move down +
a page" at 1 1
    draw cg4100a7 with all at 3
    edit var return esc pgup pgdn
  endif
  if pg eq 5 then
    *(enter boarding violation/unsafe condition information)
    if review eq N and backup eq N then
      set var pgcount to 5
    endif
    write "Press [ESC] to quit, [PGUP] to move up, [PGDN] to move +
down a page" +
    at 1 1
    draw cg4100a8 with all at 3
    edit var return esc pgup pgdn
  endif
  if pg eq 6 then
    *(enter boarding officer information)
    if review eq N and backup eq N then
      set var pgcount to 6
      set var vbounit text
      set var vbounit to "CG Group Monterey"
      set var vbodist text
      set var vbodist to 11
      set var vboopfac text
      set var vboopfac to 36268
      set var vbosrise to 06:00:00
      set var vbosset to 19:00:00
      set var vweapons to 0
    endif
    write "Press [ESC] to quit or [PGUP] to move up a page" at 1 1
    draw cg4100a9 with all at 3
    edit var return esc pgup
    set var loadmenu to "Y"
  endif
  if #return eq pgup then
    set var backup to Y
    if pg gt 1 then
      set var pg to .pg - 1
    else
      set var pg to 1
    endif
  endif
  if #return eq pgdn then
    if pg lt 6 then
      set var pg to .pg + 1
      if pgcount lt .pg then
        set var backup to N
      endif
    else
      set var pg to 6
    endif
  endif
  if #return eq esc then
    set var pg to 0
  endif
endwhile  *(end of main loop)
newpage
write "Program loading in progress.  Please stand by ........" at 10 10
if loadmenu eq "Y" then
  set var choice to 0
  while choice lt 1 or choice gt 3 then
    newpage
    write "Data Load Options" at 2 20
    write "-----------------" at 3 20
    write "Enter:" at 5 20
    write "1-To load the data into the database" at 6 20
    write "2-To ignore the data entered" at 7 20
    write "3-To go back and verify data entered" at 8 20
    write "Your choice: " at 10 20
    fillin choice using " " at 10 33
    newpage
    if choice eq 1 then
      *(test for required information)
      set var test to "F"
      if vbodate exists and vbotime exists and vbono exists and +
      vvslname exists and vobslat exists and vobslong exists and +
      vvslhin exists then
       if vviol54 exists or vviol55 exists or vviol56 exists or +
       vviol57 exists or vviol58 exists or vviol59 exists or vviol60 +
       exists or vviol61 exists or vviol62 exists or vviol63 exists then
         set var test to "T"
       endif
       if vviol64 exists or vviol65 exists or vviol66 exists or +
       vviol67 exists or vviol68 exists or vuns69 exists or vuns70 +
       exists or vuns71 exists or vuns72 exists or vuns73 exists then
         set var test to "T"
       endif
       if vuns74 exists or vuns75 exists or vuns76 exists or vuns77 +
       exists or vuns78 exists or vuns79 exists then
         set var test to "T"
       endif
       if test eq "T" then
         set error messages off
         if e1 eq 0 then
           *(vessel exists in database)
           write "changing values in vessel"
           delete rows from #1
         else
           write "loading vessel"
  set var voffno text
  set var vhome text
  set var vcall text
  set var vvalue dollar
  set var vvalue to $0
  set var vsuper text
  set var vmasts text
  set var vsailno text
  set var vhlcolor to unk
  set var vtrcolor to unk
  set var vslcolor to unk
  set var vcbcolor to unk
  set var vdkcolor to unk
  set var vmarks text
endif
load vessel
  .vvslno .vvslname .vvslhin .vvslmake .vvslmod .vvslyear +
  .vvslton .vvslft .vvslin .vvslhp .vvsluse .vvsltype +
  .vvslprop .vvslhull .vvsleng .vvslfuel .vconstr .vdecal +
  .vcmeyear .voffno .vhome .vcall .vvalue .vsuper .vmasts +
  .vsailno .vhlcolor .vtrcolor .vslcolor .vcbcolor .vdkcolor +
  .vmarks
end
if el eq 0 then
  set pointer #1 el for vessel whe vslhin eq .v3
  if el ne 0 then
    load vessel
      .v1 .v2 .v3 .v4 .v5 .v6 .v7 .v8 .v9 .v10 .v11 .v12 +
      .v13 .v14 .v15 .v16 .v17 .v18 .v19 .v20 .v21 .v22 .v23 +
      .v24 .v25 .v26 .v27 .v28 .v29 .v30 .v31 .v32
    end
  endif
endif
write "loading owvsl"
load owvsl
  .vvslhin .vownid
end
write "loading opvsl"
load opvsl
  .vvslhin .vopid
end
if vostatus eq 1 then
  load opown
    .vownid .vopid
  end
endif
set pointer #1 el for boarding where bono = .vbono
if el eq 0 then
  set var difbo to "F"
  set var b1 to bodate in #1
  set var b2 to botime in #1
  set var b3 to obslat in #1
  set var b4 to obslong in #1
  if b1 ne .vbodate or b2 ne .vbotime or b3 ne .vobslat or +
  b4 ne .vobslong then
    set var difbo to "T"
  endif
else
  set var difbo to "T"
endif
if difbo eq T then
  write "loading boarding"
  load boarding
    .vbodate .vbotime .vbono .vvslpob .vadult .vchild +
    .vostatus .vobsbody .vobcount .vobscity .vobstate +
    .vobslat .vobslong .vviol54 .vviol55 .vviol56 .vviol57 +
    .vviol58 .vviol59 .vviol60 .vviol61 .vviol62 .vviol63 +
    .vviol64 .vviol65 .vviol66 .vviol67 .vviol68 .vuns69 +
    .vuns70 .vuns71 .vuns72 .vuns73 .vuns74 .vuns75 .vuns76 +
    .vuns77 .vuns78 .vuns79 .vborem1 .vborem2 .vborem3 +
        .vboname .vborate .vbounit .vbodist .vboopfac .vboday +
        .vbosrise .vbosset .vvslhin .vweapons
      end
    endif
    if e2 eq 0 then
      *(owner exists in database)
      write "changing values in owner"
      delete rows from #2
    else
      write "loading owner"
    endif
    load owners
      .vowlname .vowfname .vowmi .vowaddr .vowcity .vowst .vowzip +
      .vowtelno .vowdob .vownid
    end
    if e2 eq 0 then
      set pointer #2 e2 for owners whe ownid eq .ow10
      if e2 ne 0 then
        load owners
          .ow1 .ow2 .ow3 .ow4 .ow5 .ow6 .ow7 .ow8 .ow9 .ow10
        end
      endif
    endif
    if e3 eq 0 then
      *(operator exists in database)
      write "changing values in operator"
      delete rows from #3
    else
      write "loading operator"
    endif
    load operator
      .voplname .vopfname .vopmi .vopaddr .vopcity .vopst .vopzip +
      .voptelno .vopdob .vopcrse .vopid
    end
    if e3 eq 0 then
      set pointer #3 e3 for operator where opid eq .op11
      if e3 ne 0 then
        load operator
          .op1 .op2 .op2 .op3 .op4 .op5 .op6 .op7 .op8 .op9 .op10 +
          .op11
        end
      endif
    endif
    set error messages on
    write "  "
    *(return option in case of entry/edit errors)
    write "Return to correct data entered? (Y)" at 20 10
    write "Note:  correct only errors indicated by messages" +
    at 22 5
    fillin er using "(" at 20 42
    if er fails or er eq "Y" then
      set var review to "Y"
      set var pg to 1
      set pointer #1 e1 for vessel where vslhin eq .vvslhin
      set pointer #2 e2 for owners where ownid eq .vownid
      set pointer #3 e3 for operator where opid eq .vopid
      break
    else
      newpage
      write "Removing duplicate entries from tables" at 10 10
      write "Please stand by................." at 15 12
      delete duplicates from vessel
      delete duplicates from owvsl
      delete duplicates from opvsl
      delete duplicates from opown
      delete duplicates from owners
      delete duplicates from operator
    endif
  endif
else
```

```
       newpage
       write "Load not performed.  Required information missing." +
       at 5 20
       write "Required information includes:" at 10 30
       write "Boarding Date, Boarding Time, Boarding Number" at 12 35
       write "Vessel Name, Boarding Latitude, Boarding Longitude" +
       at 13 35
       write "Vessel HIN, A Violation or Unsafe Condition" at 14 35
       fillin go using "Press [ENTER] to return to data entry" at 16 25
       set var pg to 1
       set var review to Y
       if vvslhin fails then
         set var fixhin to Y
       endif
       break
     endif
    endif
    if choice eq 1 or choice eq 2 then
      *(enter additional owners)
      newpage
      write "Enter additional owners? (N)" at 12 20
      fillin addl_own using "(" at 12 45
      if addl_own exists and addl_own eq Y then
        set var more to "Y"
        while more eq Y then
          set var vowlname to unk
          set var vowfname to unk
          newpage
          write "Press [PGDN] after entering initial information to +
check if" at 1 5
          write "owner exists in database.  Press [ESC] to quit" +
          at 2 10
          draw cg4100a3 with all at 4
          edit var return esc pgdn
          if #return ne esc then
            while vowlname exists and vowfname exists then
              set pointer #2 e2 for owners where ownlname = .vowlname +
              and ownfname = .vowfname
              newpage
              while e2 eq 0 then
                *(owner last/first name exist in database)
                set var vowmi to ownmi in #2
                set var vowdob to owndob in #2
                set var vowaddr to ownaddr in #2
                set var vowcity to owncity in #2
                set var vowst to ownstate in #2
                set var vowzip to ownzip in #2
                set var vowtelno to owntelno in #2
                newpage
                draw cg4100a3 with all at 3
                draw cg4100a4 with all at 8
                write "Correct owner? (N)" at 21 20
                fillin owfound using "(" at 21 35
                if owfound exists and owfound eq Y then
                  set var ow1 to .vowlname
                  set var ow2 to .vowfname
                  set var ow3 to ownmi in #2
                  set var ow4 to ownaddr in #2
                  set var ow5 to owncity in #2
                  set var ow6 to ownstate in #2
                  set var ow7 to ownzip in #2
                  set var ow8 to owntelno in #2
                  set var ow9 to owndob in #2
                  set var ow10 to ownid in #2
                  set var vownid to ownid in #2
                  break
                endif
                next #2 e2
              endwhile
              if e2 ne 0 then
      newpage
      write "This owner does not exist in the database" at 5 5
      fillin go using "Press [ENTER] to continue" at 9 5
      clear vowmi
      clear vowaddr
      clear vowcity
      set var vowst to CA
      clear vowzip
      clear vowtelno
      set var vowdob to 01/01/1901
      compute vmaxown as max ownid from owners
      if vmaxown fails then
        set var vmaxown to 0
      endif
      set var vownid to .vmaxown + 1
    endif
    break
  endwhile
  label redoown
  newpage
  write "Press [PGDN] to load data or [ESC] to quit" at 1 1
  draw cg4100a3 with all at 3
  draw sarform5 with all at 8
  edit var return esc pgup pgdn
  if #return eq pgdn then
    newpage
    set error messages off
    if e2 eq 0 then
      *(owner exists in database)
      write "changing values in owner"
      delete rows from #2
    else
      write "loading owner"
    endif
    load owners
      .vowlname .vowfname .vowmi .vowaddr .vowcity .vowst +
      .vowzip .vowtelno .vowdob .vownid
    end
    if e2 eq 0 then
      set pointer #2 e2 for owners where ownid eq .ow10
      if e2 ne 0 then
         load owners
         .ow1 .ow2 .ow3 .ow4 .ow5 .ow6 .ow7 .ow8 .ow9 .ow10
        end
      endif
    endif
    set error messages on
    write "Return to correct errors? (N)" at 15 10
    write "Note:  correct only errors indicated by messages" +
    at 20 5
    fillin fixown using "(" at 15 36
    if fixown exists and fixown eq Y then
      set pointer #2 e2 for owners where ownid eq .vownid
      goto redoown
    endif
    set pointer #2 e2 for owvsl where vhinown eq .vvslhin +
    and idowner eq .vownid
    *(check if record already exists)
    newpage
    if e2 ne 0 then
      load owvsl
        .vvslhin .vownid
      end
    endif
  endif
endif
write "More owners to enter? (N)" at 20 20
fillin mrownrs using "(" at 20 42
if mrownrs fails or mrownrs eq N then
  set var more to N
```

```
     endif
    endwhile
   endif
   *(enter additional operators)
   newpage
   write "Enter additional operators? (N)" at 12 20
   fillin addl_opr using "(" at 12 48
   if addl_opr exists and addl_opr eq Y then
    set var more to "Y"
    while more eq Y then
     set var voplname to unk
     set var vopfname to unk
     newpage
     write '.ress [PGDN] after entering initial information to +
check if" at 1 5
     write "operator exists in database.  Press [ESC] to quit" +
     at 2 10
     draw cg4100a5 with all at 4
     edit var return esc pgdn
     if #return ne esc then
      while voplname exists and vopfname exists then
       set pointer #3 e3 for operator where oplname = .voplname +
       and opfname = .vopfname
       newpage
       while e3 eq 0 then
        *(operator last/first name exist in database)
        set var vopmi to opmi in #3
        set var vopdob to opdob in #3
        set var vopaddr to opaddr in #3
        set var vopcity to opcity in #3
        set var vopst to opstate in #3
        set var vopzip to opzip in #3
        set var voptelno to optelno in #3
        set var vopcrse to opcourse in #3
        newpage
        draw cg4100a5 with all at 3
        draw cg4100a6 with all at 8
        write "Correct operator? (N)" at 21 20
        fillin opfound using "(" at 21 38
        if opfound exists and opfound eq Y then
         set var op1 to .voplname
         set var op2 to .vopfname
         set var op3 to opmi in #3
         set var op4 to opaddr in #3
         set var op5 to opcity in #3
         set var op6 to opstate in #3
         set var op7 to opzip in #3
         set var op8 to optelno in #3
         set var op9 to opdob in #3
         set var op10 to opcourse in #3
         set var op11 to opid in #3
         set var vopid to opid in #3
         break
        endif
        next #3 e3
       endwhile
       if e3 ne 0 then
        newpage
        write "This operator does not exist in the database" +
        at 5 5
        fillin go using "Press [ENTER] to continue" at 9 5
        clear vopmi
        clear vopaddr
        clear vopcity
        set var vopst to CA
        clear vopzip
        clear voptelno
        clear vopcrse
        set var vopdob to 01/01/1901
        compute vmaxoper as max opid from operator
```

```
          if vmaxoper fails then
            set var vmaxoper to 0
          endif
          set var vopid to .vmaxoper + 1
        endif
        break
      endwhile
      label redooper
      newpage
      write "Press [PGDN] to load data or [ESC] to quit" at 1 1
      draw cg4100a5 with all at 3
      draw cg4100a6 with all at 8
      edit var return esc pgup pgdn
      if #return eq pgdn then
        newpage
        set error messages off
        if e3 eq 0 then
          *(operator exists in database)
          write "changing values in operator"
          delete rows from #3
        else
          write "loading operator"
        endif
        load operator
          .voplname .vopfname .vopmi .vopaddr .vopcity .vopst +
          .vopzip .voptelno .vopdob .vopcrse .vopid
        end
        if e3 eq 0 then
          set pointer #3 e3 for operator where opid eq .op11
          if e3 ne 0 then
            load operator
              .op1 .op2 .op2 .op3 .op4 .op5 .op6 .op7 .op8 .op9 +
              .op10 .op11
            end
          endif
        endif
        set error messages on
        write "Return to correct errors? (N)" at 15 10
        write "Note:  correct only errors indicated by messages" +
        at 20 5
        fillin fixopr using "(" at 15 36
        if fixopr exists and fixopr eq Y then
          set pointer #3 e3 for operator where opid eq .vopid
          goto redooper
        endif
        set pointer #3 e3 for opvsl where vhinoper eq .vvslhin +
        and idoper eq .vopid
        *(check if record already exists)
        newpage
        if e3 ne C then
          load opvsl
            .vvslhin .vopid
          end
        endif
      endif
    endif
    write "More operators to enter? (N)" at 20 20
    fillin mropers using "(" at 20 45
    if mropers fails or mropers eq N then
      set var more to N
    endif
  endwhile
endif
newpage
write "Enter another boarding? (Y)" at 12 20
fillin cont using "(" at 12 44
if cont fails or cont eq Y then
  *(reset variables to start condition)
  clear all variables
  set var pg to 1
```

```
    set var review to "N"
    set var loadmenu to "N"
    break
   else
    set null -0-
    set var pg to 0
   endif
  endif
  if choice eq 3 then
   set var review to "Y"
   set var pg to 1
  endif
 endwhile
else
 newpage
 write "Enter another boarding? (Y)" at 12 20
 fillin cont using "(" at 12 44
 if cont fails or cont eq Y then
  *(reset variables to start condition)
  clear all variables
  set var pg to 1
  set var review to "N"
  set var loadmenu to "N"
 else
  set error messages off
  set null -0-
  set var pg to 0
 endif
endif
if pg eq 1 then
 goto begin
endif
return
*( end )


$COMMAND
ADD_OWNR
*(************************************************************************

  Program:         Add Owner
  Author:          H R Lang and J D Allen
  Date Written:    July 1987
  Description:     This program allows the user to input
                   additional owner information.
  Tables Used:     vessel, owners, owvsl
  Temp Tables:     none
  Reports Used:    none
  Forms Used:      ownrmod1, ownrmod2

************************************************************************)

set error messages on
set messages off
set escape off
newpage
write "Program loading in progress.  Please stand by ........" at 10 10
clear all variables
set null ?
set var vvslhin text
set var more to Y
while more eq Y then  *(main loop)
 set var done to Y
 newpage
 write "Enter HIN of vessel owned: " at 10 10
 fillin vvslhin using " " at 10 36
 if vvslhin exists then
  set pointer #1 e1 for vessel where vslhin eq .vvslhin
```

```
    if e1 eq 0 then
      newpage
      write "Press [PGDN] after entering initial information to check +
if" at 1 5
      write "owner exists in database.  Press [ESC] to quit" at 2 10
      draw ownrmod1 with all at 4
      edit var return esc pgdn
      if #return eq pgdn then
        if vowlname exists and vowfname exists then
          set pointer #2 e2 for owners where ownlname = .vowlname and +
          ownfname = .vowfname
          while e2 eq 0 then
            *(owner last/first name exist in database)
            set var vowmi to ownmi in #2
            set var vowdob to owndob in #2
            set var vowaddr to ownaddr in #2
            set var vowcity to owncity in #2
            set var vowst to ownstate in #2
            set var vowzip to ownzip in #2
            set var vowtelno to owntelno in #2
            newpage
            draw ownrmod1 with all at 3
            draw ownrmod2 with all at 8
            write "Correct owner? (N)" at 21 20
            fillin owfound using "(" at 21 35
            if owfound exists and owfound eq Y then
              newpage
              write "This owner already exists in the database" at 10 10
              set var vownid to ownid in #2
              set pointer #3 e3 for owvsl where vhinown eq .vvslhin +
              and idowner eq .vownid
              if e3 ne 0 then
                load owvsl
                  .vvslhin .vownid
                end
                write "The new owner-vessel record has been created" +
                at 15 10
              endif
              break
            endif
            next #2 e2
          endwhile
          if e2 ne 0 then
            clear vowmi
            clear vowaddr
            clear vowcity
            set var vowst to CA
            clear vowzip
            clear vowtelno
            set var vowdob to 01/01/1901
            clear vostatus
            compute vmaxown as max ownid from owners
            if vmaxown fails then
              set var vmaxown to 0
            endif
            set var vownid to .vmaxown + 1
            newpage
            write "Press [PGDN] to load data or [ESC] to quit" at 1 1
            draw ownrmod1 with all at 3
            draw ownrmod2 with all at 8
            edit var return esc pgdn
            if #return eq pgdn then
              set var out to N
              set var loadok to N
              while loadok eq N then
                newpage
                load owners
                  .vowlname .vowfname .vowmi .vowaddr .vowcity .vowst +
                  .vowzip .vowtelno .vowdob .vownid
                end
                write "Return to correct errors? (N)" at 10 10
                write "Note:  correct only errors indicated by +
messages" at 15 5
                fillin er using "(" at 10 36
                if er fails or er eq N then
                  set var loadok to Y
                  set var done to Y
                else
                  *(edit record again)
                  newpage
                  write "Press [PGDN] to load data or [ESC] to quit" +
                  at 1 1
                  draw ownrmod1 with all at 3
                  draw ownrmod2 with all at 8
                  edit var return esc pgdn
                  if #return eq esc then
                    set var out to Y
                    break
                  endif
                endif
              endwhile
              if out eq N then
                load owvsl
                  .vvslhin .vownid
                end
              endif
            endif
          endif
        endif
      endif
    else
      newpage
      write "This HIN does not exist in the database" at 10 10
      write "Do you wish to reenter the HIN? (Y)" at 15 10
      fillin reenter using "(" at 15 42
      if reenter fails or reenter eq Y then
        set var done to N
      endif
    endif
  endif
  if done eq Y then
    newpage
    write "More owners to be entered? (N)" at 12 20
    fillin mrownrs using "(" at 12 47
    if mrownrs fails or mrownrs eq N then
      set var more to N
    else
      clear all variables
      set var more to Y
    endif
  endif
endwhile  *(end main loop)
set error messages on
set null -0-
return
*( end )
```

APPENDIX H
MODIFY SOURCE CODE

```
$COMMAND
MODMAIN
*(**********************************************************************

                      US COAST GUARD GROUP MONTEREY
         SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                      INTELLIGENCE SYSTEM (SARELTIS)

                    MODIFY MAIN MENU and APPLICATIONS


                 Author:    Jon D. Allen, LT, USCG
                            Heidi R. Lang, LT, USN
                            AUGUST 1987

**********************************************************************)

set messages off
set escape off
newpage
clear all variables
set var pick3 integer
newpage
choose pick3 from mod_menu  in modify.apx
if pick3 eq 0 then
   quit to restmod.cmd
endif
if pick3 eq 1 then
   run mod_sar in modify.apx
   quit to restmod.cmd
endif
if pick3 eq 2 then
   run mod_brd in modify.apx
   quit to restmod.cmd
endif
if pick3 eq 3 then
   run mod_vsl in modify.apx
   quit to restmod.cmd
endif
if pick3 eq 4 then
   run mod_owop in modify.apx
   quit to restmod.cmd
endif
if pick3 eq 5 then
   run mod_sut in modify.apx
   quit to restmod.cmd
endif
if pick3 eq 6 then
   run am_vint in modify.apx
   quit to restmod.cmd
endif
if pick3 eq 7 then
   run am_pint in modify.apx
   quit to restmod.cmd
endif
if pick3 eq 8 then
   run mod_vhin in modify.apx
   quit to restmod.cmd
endif
if pick3 eq 9 then
```

```
   quit to restmain.cmd
endif
quit to restmod.cmd
return
$menu
mod_menu
column SARELTIS --- MODIFY MENU
MODIFY SAR INFORMATION
MODIFY BOARDING INFORMATION
MODIFY VESSEL INFORMATION
MODIFY OWNER/OPERATOR INFORMATION
MODIFY SAR UNIT INFORMATION
MODIFY VESSEL INTELLIGENCE
MODIFY PERSONNEL INTELLIGENCE
MODIFY VESSEL HIN
RETURN TO MAIN MENU
*( end )


$COMMAND
MOD_SAR
*(***********************************************************************

    Program:             Modify Sar
    Author:              H R Lang and J D Allen
    Date Written:        August 1987
    Description:         This program allows the user to modify
                         sar information in the database.
    Tables Used:         sar, vessel
    Temp Tables:         none
    Reports Used:        none
    Forms Used:          sarmod

************************************************************************)

set error messages on
set messages off
set escape off
newpage
write "Program loading in progress.  Please stand by ......." at 10 10
clear all variables
set null ?
set var more to Y
set var vfoldno text
while more eq Y then
  set var change to N
  set var reenter to N
  newpage
  write "Enter folder number of sar case requiring modification: " +
  at 10 5
  fillin vfoldno using " " at 10 60
  while vfoldno exists then
    newpage
    write "Searching SAR Table.  Please stand by ...." at 10 10
    set pointer #1 el for sar where folderno eq .vfoldno
    if el eq 0 then
      *(sar case exists in database)
      set var vucn to ucn in #1
      set var vmucn to mucn in #1
      set var vsardate to sardate in #1
      set var vsarday to sarday in #1
      set var vsartime to sartime in #1
      set var vsunrise to ssunrise in #1
      set var vsunset to ssunset in #1
      set var vsarnod to sarnod in #1
      set var vsarpob to sarpob in #1
      set var vsarhin to sarvhin in #1
      set var vsarlat to sarlat in #1
    set var vsarlong to sarlong in #1
    set var vdistoff to offshore in #1
    set var vimmed to immedsar in #1
    set var vwinddir to winddir in #1
    set var vwindspd to windspd in #1
    set var vvisdist to visdist in #1
    set var vclouds to clouds in #1
    set var vceiling to ceiling in #1
    set var vseas to seas in #1
    set var vswells to swells in #1
    set var oldfold to .vfoldno
    set var change to Y
    set var s1 to folderno in #1
    set var s2 to ucn in #1
    set var s3 to mucn in #1
    set var s4 to sardate in #1
    set var s5 to sarday in #1
    set var s6 to sartime in #1
    set var s7 to ssunrise in #1
    set var s8 to ssunset in #1
    set var s9 to sarnod in #1
    set var s10 to sarpob in #1
    set var s11 to sarvhin in #1
    set var s12 to sarlat in #1
    set var s13 to sarlong in #1
    set var s14 to offshore in #1
    set var s15 to immedsar in #1
    set var s16 to winddir in #1
    set var s17 to windspd in #1
    set var s18 to visdist in #1
    set var s19 to clouds in #1
    set var s20 to ceiling in #1
    set var s21 to seas in #1
    set var s22 to swells in #1
  else
    newpage
    write "This folder number does not exist in the database" at 10 10
    write "Do you wish to reenter the folder number? (Y)" at 12 10
    fillin correct using "(" at 12 52
    if correct fails or correct eq Y then
      set var reenter to Y
    endif
  endif
  break
endwhile
while change eq Y then
  set pointer #2 e2 for vessel where vslhin eq .vsarhin
  set var vvslname to vslname in #2
  newpage
  write "press [PGDN] to load changes or [ESC] to quit" at 1 1
  draw sarmod with all at 3
  edit var vfoldno vsardate vucn vmucn vsarday vsartime vsarpob +
  vsarnod vimmed vsarlat vsarlong vdistoff vvisdist vsunrise vsunset +
  vceiling vclouds vwinddir vwindspd vseas vswells return esc pgdn
  if #return eq pgdn then
    newpage
    write "Changing values in sar"
    if oldfold ne .vfoldno then
      set pointer #2 e2 for sarunits where foldno eq .oldfold
      while e2 eq 0 then
        change foldno to .vfoldno in #2
        next #2 e2
      endwhile
    endif
    delete rows from #1
    load sar
      .vfoldno .vucn .vmucn .vsardate .vsarday .vsartime .vsunrise +
      .vsunset .vsarnod .vsarpob .vsarhin .vsarlat .vsarlong +
      .vdistoff .vimmed .vwinddir .vwindspd .vvisdist .vclouds +
      .vceiling .vseas .vswells
      end
      set pointer #2 e2 for sar where folderno eq .s1 or folderno eq +
      .vfoldno
      if e2 ne 0 then
        load sar
          .s1 .s2 .s3 .s4 .s5 .s6 .s7 .s8 .s9 .s10 .s11 .s12 .s13 .s14 +
          .s15 .s16 .s17 .s18 .s19 .s20 .s21 .s22
        end
        set pointer #3 e3 for sarunits where foldno eq .vfoldno
        while e3 eq 0
          change foldno to .s1 in #3
          next #3 e3
        endwhile
      endif
      write "Return to correct data entered? (N)" at 15 10
      write "Note:  correct only errors indicated by messages" at 20 5
      fillin er using "(" at 15 42
      if er fails or er eq N then
        set var change to N
      endif
    else
      set var change to N
    endif
  endwhile
  if reenter eq N then
    newpage
    write "More sar cases to be changed? (N)" at 12 20
    fillin mrsar using "(" at 12 50
    if mrsar fails or mrsar eq N then
      set var more to N
    endif
  endif
endwhile  *(end of main loop)
set error messages off
set null -0-
return
*( end )


$COMMAND
MOD_BRD
*(*********************************************************************

    Program:           Modify Boarding
    Author:            H R Lang and J D Allen
    Date Written:      August 1987
    Description:       This program allows the user to modify
                       boarding information in the database.
    Tables Used:       boarding, vessel
    Temp Tables:       none
    Reports Used:      none
    Forms Used:        bomod1, bomod2, bomod3

*********************************************************************)

set error messages on
set messages off
set escape off
newpage
write "Program loading in progress.  Please stand by ......." at 10 10
clear all variables
set null ?
set var pg to 1
set var review to N
*(controls relation search- if reviewing, don't search)
while pg gt 0 then  *(main loop)
  if review eq N then
    newpage
```

```
set var fix to N
set var chk-mr to N
set var vbono text
write "Enter report number of boarding requiring modification: " +
at 10 5
fillin vbono using " " at 10 60
while vbono exists then
  newpage
  write "Searching Boarding Table.  Please stand by ......" at 10 10
  set pointer #1 e1 for boarding where bono eq .vbono
  if e1 eq 0 then
    *(record exists in database)
    set var vbodate to bodate in #1
    set var vbotime to botime in #1
    set var vvslpob to vslpob in #1
    set var vadult to adultpfd in #1
    set var vchild to childpfd in #1
    set var vostatus to owstatus in #1
    set var vobsbody to obsbody in #1
    set var vobcount to obscount in #1
    set var vobscity to obscity in #1
    set var vobstate to obsstate in #1
    set var vobslat to obslat in #1
    set var vobslong to obslong in #1
    set var vviol54 to viol54 in #1
    set var vviol55 to viol55 in #1
    set var vviol56 to viol56 in #1
    set var vviol57 to viol57 in #1
    set var vviol58 to viol58 in #1
    set var vviol59 to viol59 in #1
    set var vviol60 to viol60 in #1
    set var vviol61 to viol61 in #1
    set var vviol62 to viol62 in #1
    set var vviol63 to viol63 in #1
    set var vviol64 to viol64 in #1
    set var vviol65 to viol65 in #1
    set var vviol66 to viol66 in #1
    set var vviol67 to viol67 in #1
    set var vviol68 to viol68 in #1
    set var vuns69 to uns69 in #1
    set var vuns70 to uns70 in #1
    set var vuns71 to uns71 in #1
    set var vuns72 to uns72 in #1
    set var vuns73 to uns73 in #1
    set var vuns74 to uns74 in #1
    set var vuns75 to uns75 in #1
    set var vuns76 to uns76 in #1
    set var vuns77 to uns77 in #1
    set var vuns78 to uns78 in #1
    set var vuns79 to uns79 in #1
    set var vborem1 to borem1 in #1
    set var vborem2 to borem2 in #1
    set var vborem3 to borem3 in #1
    set var vboname to boname in #1
    set var vborate to borate in #1
    set var vbounit to bounit in #1
    set var vbodist to bodist in #1
    set var vboopfac to boopfac in #1
    set var vboday to boday in #1
    set var vbosrise to bsunrise in #1
    set var vbosset to bsunset in #1
    set var vvslhin to bovhin in #1
    set var vweapons to weapons in #1
    set var b1 to bodate in #1
    set var b2 to botime in #1
    set var b3 to bono in #1
    set var b4 to vslpob in #1
    set var b5 to adultpfd in #1
    set var b6 to childpfd in #1
    set var b7 to owstatus in #1
```

```
        set var b8 to obsbody in #1
        set var b9 to obscount in #1
        set var b10 to obscity in #1
        set var b11 to obsstate in #1
        set var b12 to obslat in #1
        set var b13 to obslong in #1
        set var b14 to viol54 in #1
        set var b15 to viol55 in #1
        set var b16 to viol56 in #1
        set var b17 to viol57 in #1
        set var b18 to viol58 in #1
        set var b19 to viol59 in #1
        set var b20 to viol60 in #1
        set var b21 to viol61 in #1
        set var b22 to viol62 in #1
        set var b23 to viol63 in #1
        set var b24 to viol64 in #1
        set var b25 to viol65 in #1
        set var b26 to viol66 in #1
        set var b27 to viol67 in #1
        set var b28 to viol68 in #1
        set var b29 to uns69 in #1
        set var b30 to uns70 in #1
        set var b31 to uns71 in #1
        set var b32 to uns72 in #1
        set var b33 to uns73 in #1
        set var b34 to uns74 in #1
        set var b35 to uns75 in #1
        set var b36 to uns76 in #1
        set var b37 to uns77 in #1
        set var b38 to uns78 in #1
        set var b39 to uns79 in #1
        set var b40 to borem1 in #1
        set var b41 to borem2 in #1
        set var b42 to borem3 in #1
        set var b43 to boname in #1
        set var b44 to borate in #1
        set var b45 to bounit in #1
        set var b46 to bodist in #1
        set var b47 to boopfac in #1
        set var b48 to boday in #1
        set var b49 to bsunrise in #1
        set var b50 to bsunset in #1
        set var b51 to bovhin in #1
        set var b52 to weapons in #1
        set pointer #2 e2 for vessel where vslhin eq .vvslhin
        *(get vessel name)
        set var vvslname to vslname in #2
        set var review to Y
        break
      else
        newpage
        write "This boarding report number does not exist in the +
database" at 10 10
        write "Do you wish to reenter the report number? (Y)" at 12 10
        fillin correct using "(" at 12 52
        if correct exists and correct eq N then
          set var chk-mr to Y
          set var fix to Y
          break
        else
          set var fix to Y
          break
        endif
      endif
    endwhile
  endif
  if fix eq N then
    if pg eq 1 then
      newpage
      write "Press [PGDN] to move down a page or [ESC] to quit" at 1 1
      draw bomod1 with all at 3
      edit var vbono vbodate vbotime vboday vvslpob vadult vchild +
      vostatus vobsbody vobcount vobscity vobstate vobslat vobslong +
      return esc pgdn
    endif
    if pg eq 2 then
      newpage
      write "Press [ESC] to quit, [PGUP] to move up or [PGDN] to move +
down a page" at 1 1
      draw bomod2 with all at 3
      edit var return esc pgup pgdn
    endif
    if pg eq 3 then
      newpage
      write "Press [PGUP] to move up a page or [ESC] to quit" at 1 1
      draw bomod3 with all at 3
      edit var return esc pgup
    endif
    if #return eq pgup then
      if pg gt 1 then
        set var pg to .pg - 1
      else
        set var pg to 1
      endif
    endif
    if #return eq pgdn then
      if pg lt 3 then
        set var pg to .pg + 1
      else
        set var pg to 3
      endif
    endif
    if #return eq esc then
      set var choice to 0
      while choice lt 1 or choice gt 3 then
        newpage
        write  "Boarding Change Options" at 2 20
        write  "-----------------------" at 3 20
        write  "1.  Change values in the database" at 5 20
        write  "2.  Go back and verify data " at 6 20
        write  "3.  Quit " at 7 20
        write  "Your choice: " at 9 25
        fillin choice using " " at 9 37
        if choice eq 1 then
          set error messages off
          newpage
          write "changing values in boarding"
          delete rows from #1
          load boarding
            .vbodate .vbotime .vbono .vvslpob .vadult .vchild +
            .vostatus .vobsbody .vobcount .vobscity .vobstate +
            .vobslat .vobslong .vviol54 .vviol55 .vviol56 .vviol57 +
            .vviol58 .vviol59 .vviol60 .vviol61 .vviol62 .vviol63 +
            .vviol64 .vviol65 .vviol66 .vviol67 .vviol68 .vuns69 +
            .vuns70 .vuns71 .vuns72 .vuns73 .vuns74 .vuns75 .vuns76
            .vuns77 .vuns78 .vuns79 .vborem1 .vborem2 .vborem3 +
            .vboname .vborate .vbounit .vbodist .vboopfac .vboday +
            .vbosrise .vbosset .vvslhin .vweapons
          end
          set pointer #3 e3 for boarding where bono eq .b3
          if e3 ne 0 then
            load boarding
              .b1 .b2 .b3 .b4 .b5 .b6 .b7 .b8 .b9 .b10 .b11 .b12 .b13 +
              .b14 .b15 .b16 .b17 .b18 .b19 .b20 .b21 .b22 .b23 .b24 +
              .b25 .b26 .b27 .b28 .b29 .b30 .b31 .b32 .b33 .b34 .b35 +
              .b36 .b37 .b38 .b39 .b40 .b41 .b42 .b43 .b44 .b45 .b46 +
              .b47 .b48 .b49 .b50 .b51 .b52
            end
          endif
          set error messages on
          newpage
          write "Return to correct data entered? (N)" at 15 10
          fillin er using "(" at 15 42
          if er exists and er eq Y then
            set var pg to 1
          else
            set var chk-mr to Y
          endif
        endif
        if choice eq 2 then
          set var pg to 1
        endif
        if choice eq 3 then
          newpage
          write "More boardings to be changed? (N)" at 12 20
          fillin more using "(" at 12 50
          if more fails or more eq N then
            set var pg to 0
          else
            set var pg to 1
            set var review to N
          endif
        endif
      endwhile
    endif
  endif
  if chk-mr eq Y then
    newpage
    write "More boardings to be changed? (N)" at 12 20
    fillin more using "(" at 12 50
    if more fails or more eq N then
      set var pg to 0
    else
      clear all variables
      set var pg to 1
      set var review to N
    endif
  endif
endwhile  *(end of main loop)
set error messages off
set null -0-
return
*( end )


$COMMAND
MOD_VSL
*(*********************************************************************

    Program:              Modify Vessel
    Author:               H R Lang and J D Allen
    Date Written:         August 1987
    Description:          This program allows the user to modify
                          vessel information in the database.
    Tables Used:          vessel
    Temp Tables:          none
    Reports Used:         none
    Forms Used:           vslmod1, vslmod2

**********************************************************************)

set error messages on
set messages off
set escape off
newpage
write "Program loading in progress.  Please stand by ......." at 10 10
clear all variables
```

```
set null ?
set var pg to 1
set var more to Y
while more eq Y then  *(main loop)
  set var chk_hin to Y
  set var chk_name to N
  while chk_hin eq Y then
    newpage
    set var vvslhin text
    write "Enter HIN of vessel requiring modification: " at 10 10
    write "(enter 'xxx' to go directly to vessel name search)" at 12 5
    fillin vvslhin using " " at 10 53
    while vvslhin exists then
      if vvslhin eq xxx then
        set var chk_hin to N
        set var chk_name to Y
        break
      endif
      newpage
      write "Searching Vessel Table.  Please stand by ......." at 10 10
      set pointer #1 e1 for vessel where vslhin = .vvslhin
      newpage
      if e1 eq 0 then
        *(vessel exists in database)
        set var vvslno to vslno in #1
        set var vvslname to vslname in #1
        set var vvslmake to vslmake in #1
        set var vvslmod to vslmodel in #1
        set var vvslyear to vslyear in #1
        set var vvslton to vsltons in #1
        set var vvslft to vsllenft in #1
        set var vvslin to vsllenin in #1
        set var vvslhp to vslhp in #1
        set var vvsluse to vsluse in #1
        set var vvsltype to vsltype in #1
        set var vvslprop to vslprop in #1
        set var vvslhull to vslhull in #1
        set var vvsleng to vsleng in #1
        set var vvslfuel to vslfuel in #1
        set var vconstr to vslconst in #1
        set var vdecal to cmedecal in #1
        set var vcmeyear to cmeyear in #1
        set var voffno to vsloffno in #1
        set var vhome to homeport in #1
        set var vcall to vslcall in #1
        set var vvalue to vslvalue in #1
        set var vsuper to vslsuper in #1
        set var vmasts to vslmasts in #1
        set var vsailno to sailno in #1
        set var vhlcolor to hlcolor in #1
        set var vtrcolor to trcolor in #1
        set var vslcolor to slcolor in #1
        set var vcbcolor to cbcolor in #1
        set var vdkcolor to dkcolor in #1
        set var vmarks to vslmarks in #1
        set var v1 to vslno in #1
        set var v2 to vslname in #1
        set var v3 to vslhin in #1
        set var v4 to vslmake in #1
        set var v5 to vslmodel in #1
        set var v6 to vslyear in #1
        set var v7 to vsltons in #1
        set var v8 to vsllenft in #1
        set var v9 to vsllenin in #1
        set var v10 to vslhp in #1
        set var v11 to vsluse in #1
        set var v12 to vsltype in #1
        set var v13 to vslprop in #1
        set var v14 to vslhull in #1
        set var v15 to vsleng in #1
      set var v16 to vslfuel in #1
      set var v17 to vslconst in #1
      set var v18 to cmedecal in #1
      set var v19 to cmeyear in #1
      set var v20 to vsloffno in #1
      set var v21 to homeport in #1
      set var v22 to vslcall in #1
      set var v23 to vslvalue in #1
      set var v24 to vslsuper in #1
      set var v25 to vslmasts in #1
      set var v26 to sailno in #1
      set var v27 to hlcolor in #1
      set var v28 to trcolor in #1
      set var v29 to slcolor in #1
      set var v30 to cbcolor in #1
      set var v31 to dkcolor in #1
      set var v32 to vslmarks in #1
      set var chk_hin to N
      break
    else
      newpage
      write "This HIN does not exist in the database" at 10 20
      write "Do you wish to reenter the HIN? (Y)" at 12 20
      fillin correct using "(" at 12 52
      if correct exists and correct eq N then
        set var chk_hin to N
        set var chk_name to Y
      endif
      break
    endif
  endwhile
endwhile
while chk_name eq Y then
  newpage
  write "Enter name of vessel requiring modification: " at 10 10
  fillin vvslname using " " at 10 54
  while vvslname exists then
    newpage
    write "Searching Vessel Table.  Please stand by ........" +
    at 10 10
    set pointer #1 e1 for vessel where vslname = .vvslname
    newpage
    while e1 eq 0 then
      *(vessel exists in database)
      set var vvslno to vslno in #1
      set var vvslname to vslname in #1
      set var vvslhin to vslhin in #1
      set var vvslmake to vslmake in #1
      set var vvslmod to vslmodel in #1
      set var vvslyear to vslyear in #1
      set var vvslton to vsltons in #1
      set var vvslft to vsllenft in #1
      set var vvslin to vsllenin in #1
      set var vvslhp to vslhp in #1
      set var vvsluse to vsluse in #1
      set var vvsltype to vsltype in #1
      set var vvslprop to vslprop in #1
      set var vvslhull to vslhull in #1
      set var vvsleng to vsleng in #1
      set var vvslfuel to vslfuel in #1
      set var vconstr to vslconst in #1
      set var vdecal to cmedecal in #1
      set var vcmeyear to cmeyear in #1
      set var voffno to vsloffno in #1
      set var vhome to homeport in #1
      set var vcall to vslcall in #1
      set var vvalue to vslvalue in #1
      set var vsuper to vslsuper in #1
      set var vmasts to vslmasts in #1
      set var vsailno to sailno in #1
      set var vhlcolor to hlcolor in #1
      set var vtrcolor to trcolor in #1
      set var vslcolor to slcolor in #1
      set var vcbcolor to cbcolor in #1
      set var vdkcolor to dkcolor in #1
      set var vmarks to vslmarks in #1
      newpage
      draw vslmodl with all at 1
      write "Correct vessel? (N)" at 22 20
      fillin vslfound using "(" at 22 36
      if vslfound exists and vslfound eq Y then
        set var chk_name to N
        set var v1 to vslno in #1
        set var v2 to vslname in #1
        set var v3 to vslhin in #1
        set var v4 to vslmake in #1
        set var v5 to vslmodel in #1
        set var v6 to vslyear in #1
        set var v7 to vsltons in #1
        set var v8 to vsllenft in #1
        set var v9 to vsllenin in #1
        set var v10 to vslhp in #1
        set var v11 to vsluse in #1
        set var v12 to vsltype in #1
        set var v13 to vslprop in #1
        set var v14 to vslhull in #1
        set var v15 to vsleng in #1
        set var v16 to vslfuel in #1
        set var v17 to vslconst in #1
        set var v18 to cmedecal in #1
        set var v19 to cmeyear in #1
        set var v20 to vsloffno in #1
        set var v21 to homeport in #1
        set var v22 to vslcall in #1
        set var v23 to vslvalue in #1
        set var v24 to vslsuper in #1
        set var v25 to vslmasts in #1
        set var v26 to sailno in #1
        set var v27 to hlcolor in #1
        set var v28 to trcolor in #1
        set var v29 to slcolor in #1
        set var v30 to cbcolor in #1
        set var v31 to dkcolor in #1
        set var v32 to vslmarks in #1
        break
      endif
      next #1 e1
    endwhile
    if e1 ne 0 then
      newpage
      write "This vessel does not exist in the database" at 10 20
      write "Do you wish to reenter the vessel name? (Y)" at 12 20
      fillin correct using "(" at 12 60
      if correct exists and correct eq N then
        set var chk_name to N
      endif
    endif
    break
  endwhile
endwhile
if e1 eq 0 then
  set var error to Y
  while error eq Y then
    set var error to N
    newpage
    write "Press [ESC] to quit or [PGDN] to move down a page" at 1 1
    draw vslmodl with all at 3
    edit var vvslname vvslno voffno vvslmake vvslmod vvslyear vvslft +
    vvslin vvslhp vvslton vvsluse vvsltype vvslprop vvslhull vvsleng +
    vvslfuel vconstr return esc pgdn
```

```
      if #return eq pgdn then
        newpage
        write "Press [ESC] to quit or [PGDN] to load changes" at 1 1
        draw vslmod2 with all at 3
        edit var vdecal vcmeyear vhome vcall vvalue vsuper vmasts +
        vslcolor vsailno vhlcolor vtrcolor vcbcolor vdkcolor vmarks +
        return esc pgdn
        if #return eq pgdn then
          *(vessel exists in database)
          set error messages off
          newpage
          write "changing values in vessel"
          delete rows from #1
          load vessel
            .vvslno .vvslname .vvslhin .vvslmake .vvslmod .vvslyear +
            .vvslton .vvslft .vvslin .vvslhp .vvsluse .vvsltype +
            .vvslprop .vvslhull .vvsleng .vvslfuel .vconstr .vdecal +
            .vcmeyear .voffno .vhome .vcall .vvalue .vsuper .vmasts +
            .vsailno .vhlcolor .vtrcolor .vslcolor .vcbcolor .vdkcolor +
            .vmarks
          end
          set pointer #2 e2 for vessel where vslhin eq .v3
          if e2 ne 0 then
            load vessel
              .v1 .v2 .v3 .v4 .v5 .v6 .v7 .v8 .v9 .v10 .v11 .v12 .v13 +
              .v14 .v15 .v16 .v17 .v18 .v19 .v20 .v21 .v22 .v23 .v24 +
              .v25 .v26 .v27 .v28 .v29 .v30 .v31 .v32
            end
          endif
          set error messages on
          write "Return to correct data entered? (N)" at 15 10
          write "Note:  correct only errors identified by messages" +
          at 20 5
          fillin er using "(" at 15 42
          if er exists and er eq Y then
            set var error to Y
          endif
        endif
      endif
    endwhile
  endif
  newpage
  write "More vessels to be changed? (N)" at 12 20
  fillin mrvsls using "(" at 12 48
  if mrvsls fails or mrvsls eq N then
    set var more to N
  endif
endwhile  *(end of main loop)
set error messages off
set null -0-
return
*( end )


$COMMAND
MOD_OWOP
*(***********************************************************************

    Prcgram:           Modify Owner and Operator
    Author:            H R Lang and J D Allen
    Date Written:      August 1987
    Description:       This program allows the user to modify
                       owner and operator information in the
                       database.
    Tables Used:       owners, operator, opown
    Temp Tables:       none
    Reports Used:      none
    Forms Used:        ownrmod1, ownrmod2, opermod1, opermod2
```

```
***********************************************************************)

set error messages on
set messages off
set escape off
newpage
write "Program loading in progress.  Please stand by ......." at 10 10
clear all variables
set null ?
set var owchange to N
set var opchange to N
set var choice to 0
while choice gt 3 or choice lt 1 then  *(main loop)
  newpage
  write "Owner / Operator Change Options" at 10 20
  write "-------------------------------" at 11 20
  write "1.  change owner values" at 13 22
  write "2.  change operator values" at 14 22
  write "3.  quit" at 15 22
  write "Your choice: " at 17 25
  fillin choice using " " at 17 37
  if choice eq 1 then
    set var chk_ownr to Y
    while chk_ownr eq Y then
      *(change owner information)
      newpage
      write "Press [PGDN] after entering initial information to check +
if" at 1 5
      write "owner exists in database.  Press [ESC] to quit" at 2 10
      draw ownrmodl at 4
      enter var return esc pgdn
      if #return ne esc then
        while vowlname exists and vowfname exists then
          set pointer #2 e2 for owners where ownlname = .vowlname and +
          ownfname = .vowfname
          newpage
          while e2 eq 0 then
            *(owner last/first name exist in database)
            set var vowmi to ownmi in #2
            set var vowdob to owndob in #2
            set var vowaddr to ownaddr in #2
            set var vowcity to owncity in #2
            set var vowst to ownstate in #2
            set var vowzip to ownzip in #2
            set var vowtelno to owntelno in #2
            newpage
            draw ownrmodl with all at 3
            draw ownrmod2 with all at 8
            write "Correct owner? (N)" at 21 20
            fillin owfound using "(" at 21 35
            if owfound exists and owfound eq Y then
              set var ow1 to .vowlname
              set var ow2 to .vowfname
              set var ow3 to ownmi in #2
              set var ow4 to ownaddr in #2
              set var ow5 to owncity in #2
              set var ow6 to ownstate in #2
              set var ow7 to ownzip in #2
              set var ow8 to owntelno in #2
              set var ow9 to owndob in #2
              set var ow10 to ownid in #2
              set var vownid to ownid in #2
              newpage
              write "Do you wish to change name? (N)" at 10 10
              fillin nmchange using "(" at 10 38
              if nmchange exists and nmchange eq Y then
                newpage
                write "Press [PGDN] to make change or [ESC] to quit" +
                at 1 1
```

```
                  draw ownrmodl with all at 3
                  edit var return esc pgdn
                  if #return eq pgdn then
                    change ownfname to .vowfname in #2
                    change ownlname to .vowlname in #2
                  endif
                  set var chk_ownr to N
                  break
                endif
                newpage
                write "Press [PGDN] to make changes or [ESC] to quit" +
                at 1 1
                draw ownrmodl with all at 3
                draw ownrmod2 with all at 8
                edit var return esc pgdn
                if #return eq pgdn then
                  set var er to Y
                  while er eq Y then
                    set error messages off
                    newpage
                    write "changing values in owner"
                    delete rows from #2
                    load owners
                      .vowlname .vowfname .vowmi .vowaddr .vowcity +
                      .vowst .vowzip .vowtelno .vowdob .vownid
                    end
                    set pointer #3 e3 for owners whe ownid eq .ow10
                    if e3 ne 0 then
                    load owners
                      .owl .ow2 .ow3 .ow4 .ow5 .ow6 .ow7 .ow8 .ow9 .ow10
                    end
                    endif
                    set error messages on
                    write "Return to make corrections? (N)" at 10 10
                    write "Note:  correct only errors identified by +
messages" at 15 5
                    fillin error using "(" at 10 38
                    if error exists and error eq Y then
                      newpage
                      write "Press [PGDN] to make changes"
                      draw ownrmodl with all at 3
                      draw ownrmod2 with all at 8
                      edit var return esc pgdn
                    else
                      set var er to N
                    endif
                  endwhile
                  set pointer #1 el for opown where ownno eq .vownid
                  newpage
                  if el eq 0 then
                    set var vopid to opno in #1
                    set error messages off
                    set pointer #1 el for operator where opid eq .vopid
                    set var vopcrse to opcourse in #1
                    delete rows from #1
                    load operator
                      .vowlname .vowfname .vowmi .vowaddr .vowcity +
                      .vowst .vowzip .vowtelno .vowdob .vopcrse .vopid
                    end
                    set pointer #1 el for operator where opid eq .vopid
                    if el ne 0 then
                      load operator
                        .owl .ow2 .ow3 .ow4 .ow5 .ow6 .ow7 .ow8 .ow9 +
                        .vopcrse .vopid
                      end
                    endif
                    set error messages on
                    write "Values for this person also changed in +
operator"
                    set var chk_ownr to N
                  set var owchange to N
                  break
                else
                  set var owchange to Y
                  if opchange eq Y then
                    if vowlname eq .voplname then
                      newpage
                      draw owopform with all at 1
                      write "Is this the same person? (N)" at 20 20
                      fillin same using "(" at 20 45
                      if same exists and same eq Y then
                        load opown
                          .vownid .vopid
                        end
                      endif
                    endif
                  endif
                  set var chk_ownr to N
                  break
                endif
              else
                break
              endif
            endif
            next #2 e2
          endwhile
          if e2 ne 0 then
            newpage
            write "This owner does not exist in the database" at 5 5
            write "Do you wish to reenter name? (Y)" at 10 5
            fillin go using "(" at 10 34
            if go exists and go eq N then
              set var chk_ownr to N
            endif
          endif
          break
        endwhile
      else
        set var chk_ownr to N
      endif
    endwhile
    set var choice to 0
  endif
  if choice eq 2 then
    set var chk_oper to Y
    while chk_oper eq Y then
      *(change operator information)
      newpage
      write "Press [PGDN] after entering initial information to check +
if" at 1 5
      write "operator exists in database.  Press [ESC] to quit" at 2 10
      draw opermodl at 4
      enter var return esc pgdn
      if #return ne esc then
        while voplname exists and vopfname exists then
          set pointer #2 e2 for operator where oplname = .voplname and +
          opfname = .vopfname
          newpage
          while e2 eq 0 then
            *(operator last/first name exist in database)
            set var vopmi to opmi in #2
            set var vopdob to opdob in #2
            set var vopaddr to opaddr in #2
            set var vopcity to opcity in #2
            set var vopst to opstate in #2
            set var vopzip to opzip in #2
            set var voptelno to optelno in #2
            set var vopcrse to opcourse in #2
            newpage
            draw opermodl with all at 3
```

```
             draw opermod2 with all at 8
             write "Correct operator? (N)" at 22 20
             fillin opfound using "(" at 22 38
             if opfound exists and opfound eq Y then
               set var op1 to .voplname
               set var op2 to .vopfname
               set var op3 to opmi in #2
               set var op4 to opaddr in #2
               set var op5 to opcity in #2
               set var op6 to opstate in #2
               set var op7 to opzip in #2
               set var op8 to optelno in #2
               set var op9 to opdob in #2
               set var op10 to opcourse in #2
               set var op11 to opid in #2
               set var vopid to opid in #2
               newpage
               write "Do you wish to change name? (N)" at 10 10
               fillin nmchange using "(" at 10 38
               if nmchange exists and nmchange eq Y then
                 newpage
                 write "Press [PGDN] to make change or [ESC] to quit" +
                 at 1 1
                 draw opermod1 with all at 3
                 edit var return esc pgdn
                 if #return eq pgdn then
                   change opfname to .vopfname in #2
                   change oplname to .voplname in #2
                 endif
                 set var chk_oper to N
                 break
               endif
               newpage
               write "Press [PGDN] to make changes or [ESC] to quit" +
               at 1 1
               draw opermod1 with all at 3
               draw opermod2 with all at 8
               edit var return esc pgdn
               if #return eq pgdn then
                 set var er to Y
                 while er eq Y then
                   set error messages off
                   newpage
                   write "changing values in operator"
                   delete rows from #2
                   load operator
                     .voplname .vopfname .vopmi .vopaddr .vopcity +
                     .vopst .vopzip .voptelno .vopdob .vopcrse .vopid
                   end
                   set pointer #3 e3 for operator where opid eq .op11
                   if e3 ne 0 then
                   load operator
                     .op1 .op2 .op2 .op3 .op4 .op5 .op6 .op7 .op8 .op9 +
                     .op10 .op11
                   end
                   endif
                   set error messages on
                   write "Return to make corrections? (N)" at 10 10
                   write "Note:  correct only errors identified by +
messages" at 15 5
                   fillin error using "(" at 10 38
                   if error exists and error eq Y then
                     newpage
                     write "Press [PGDN] to make changes"
                     draw opermod1 with all at 3
                     draw opermod2 with all at 8
                     edit var return esc pgdn
                   else
                     set var er to N
                   endif
```

```
                endwhile
                set pointer #1 e1 for opown where opno eq .vopid
                newpage
                if e1 eq 0 then
                  set var vownid to ownno in #1
                  set error messages off
                  set pointer #1 e1 for owner where ownid eq .vownid
                  change ownlname to .voplname in #1
                  change ownfname to .vopfname in #1
                  change ownmi to .vopmi in #1
                  change ownaddr to .vopaddr to #1
                  change owncity to .vopcity in #1
                  change ownstate to .vopst in #1
                  change ownzip to .vopzip in #1
                  change owntelno to .voptelno in #1
                  change owndob to .vopdob in #1
                  set error messages on
                  write "Values for this person also changed in owner"
                  set var chk_oper to N
                  set var opchange to N
                  break
                else
                  set var opchange to Y
                  if owchange eq Y then
                    if voplname eq .vowlname then
                      newpage
                      draw owopform with all at 1
                      write "Is this the same person? (N)" at 20 20
                      fillin same using "(" at 20 45
                      if same exists and same eq Y then
                        load opown
                          .vownid .vopid
                        end
                      endif
                    endif
                  endif
                  set var chk_oper to N
                  break
                endif
              else
                break
              endif
            endif
            next #2 e2
          endwhile
          if e2 ne 0 then
            newpage
            write "This operator does not exist in the database" at 5 5
            write "Do you wish to reenter name? (Y)" at 10 5
            fillin go using "(" at 10 34
            if go exists and go eq N then
              set var chk_oper to N
            endif
          endif
          break
        endwhile
      else
        set var chk_oper to N
      endif
    endwhile
    set var choice to 0
  endif
endwhile  *(end of main loop)
set error messages off
set null -0-
return
*( end )
```

```
$COMMAND
MOD_SUT
*(*********************************************************************

    Program:           Modify Sar Unit
    Author:            H R Lang and J D Allen
    Date Written:      August 1987
    Description:       This program allows the user to modify
                       sar unit data in the database.
    Tables Used:       sar, sarunits
    Temp Tables:       none
    Reports Used:      none
    Forms Used:        sunitmod

**********************************************************************)

set error messages on
set messages off
set escape off
newpage
write "Program loading in progress.  Please stand by ......." at 10 10
clear all variables
set null ?
set var more to Y
set var vfoldno text
set var vunit text
set var foldok to N
while more eq Y then  *(main loop)
  set var name to N
  set var both to Y
  set var done to N
  set var change to N
  while both eq Y then  *(check both folder number and unit name)
    newpage
    if foldok eq N then
      write "Enter sar folder number of unit data requiring +
modification: " at 10 1
      fillin vfoldno using " " at 10 61
    endif
    write "Enter name of unit requiring modification: " at 12 1
    write "(or enter 'xxx' to list units)" at 13 5
    fillin vunit using " " at 12 43
    while vfoldno exists and vunit exists then
      set pointer #1 e1 for sarunits where foldno eq .vfoldno and +
      unitname eq .vunit
      newpage
      if e1 eq 0 then
        set var vunit to unitname in #1
        set var vuwtime to uwtime in #1
        set var vuwdate to uwdate in #1
        set var voseta to oseta in #1
        set var vetadate to etadate in #1
        set var vostime to ostime in #1
        set var vdateos to osdate in #1
        set var valgtime to alngtime in #1
        set var valgdate to alngdate in #1
        set var vtowtime to towtime in #1
        set var vtowdate to towdate in #1
        set var vdvmoort to dvmoored in #1
        set var vdvmoord to dvmdate in #1
        set var vsumoort to sutmoor in #1
        set var vsumoord to sutdate in #1
        set var su1 to foldno in #1
        set var su2 to unitname in #1
        set var su3 to uwtime in #1
        set var su4 to uwdate in #1
        set var su5 to oseta in #1
        set var su6 to etadate in #1
        set var su7 to ostime in #1
        set var su8 to osdate in #1
        set var su9 to alngtime in #1
        set var su10 to alngdate in #1
        set var su11 to towtime in #1
        set var su12 to towdate in #1
        set var su13 to dvmoored in #1
        set var su14 to dvmdate in #1
        set var su15 to sutmoor in #1
        set var su16 to sutdate in #1
        set var change to Y
        set var both to N
        break
      else
        set pointer #2 e2 for sarunits where foldno eq .vfoldno
        newpage
        if e2 eq 0 then
          set var foldok to Y
          if vunit eq xxx then
            set var both to N
            set var name to Y
            break
          endif
          newpage
          write "This unit name does not exist with this folder +
number" at 10 10
          write "Do you wish to reenter the unit name? (Y)" at 12 10
          fillin correct using "(" at 12 48
          if correct exists and correct eq N then
            set var both to N
            set var name to Y
          endif
          break
        else
          newpage
          write "This folder number does not exist in the database"
          at 10 10
          write "Do you wish to reenter the folder number? (Y)" at 12 10
          fillin correct using "(" at 12 52
          if correct fails or correct eq Y then
            break
          else
            set var both to N
            set var done to Y
            break
          endif
        endif
      endif
    endwhile
  endwhile
  while name eq Y then  *(search on name with existing folder number)
    newpage
    write "The following unit names exist within the indicated folder +
number" at 1 2
    write " "
    sel unitname from sarunits where foldno eq .vfoldno
    write "Do you wish to enter/reenter the unit name? (Y)" at 20 10
    fillin reenter using "(" at 20 54
    if reenter exists and reenter eq N then
      set var done to Y
    endif
    break
  endwhile
  if change eq Y then
    newpage
    write "press [PGDN] to load changes or [ESC] to quit" at 1 1
    draw sunitmod with all at 3
    edit var vunit vuwtime vuwdate voseta vetadate vostime vdateos +
    valgtime valgdate vtowtime vtowdate vdvmoort vdvmoord vsumoort +
    vsumoord return esc pgdn
    if #return eq pgdn then
```

```
      set error messages off
      delete rows from #1
      load sarunits
        .vfoldno .vunit .vuwtime .vuwdate .voseta .vetadate .vostime +
        .vdateos .valgtime .valgdate .vtowtime .vtowdate .vdvmoort +
        .vdvmoord .vsumoort .vsumoord
      end
      set pointer #2 e2 for sarunits where foldno eq .su1 and +
      unitname eq .su2
      if e2 ne 0 then
        load sarunits
          .su1 .su2 .su3 .su4 .su5 .su6 .su7 .su8 .su9 .su10 .su11 +
          .su12 .su13 .su14 .su15 .su16
        end
      endif
      set error messages on
    endif
    set var done to Y
  endif
  if done eq Y then
    newpage
    write "More sar units to be changed? (N)" at 12 20
    fillin mrsut using "(" at 12 50
      if mrsut fails or mrsut eq N then
        set var more to N
      else
        set var foldok to N
      endif
    endif
endwhile  *(end of main loop)
set error messages off
set null -0-
return
*( end )


$COMMAND
MOD_VHIN
*(**********************************************************************

    Program:               Modify Vessel HIN
    Author:                H R Lang and J D Allen
    Date Written:          August 1987
    Description:           This program allows the user to modify
                           vessel hull identification numbers in
                           the database.
    Tables Used:           vessel, sar, boarding, owvsl, opvsl
    Temp Tables:           none
    Reports Used:          none
    Forms Used:            vhinform

***********************************************************************)

set error messages on
set messages off
set escape off
newpage
write "Program loading in progress.  Please stand by ......." at 10 10
clear all variables
set null ?
set var more to Y
while more eq Y then  *(main loop)
  set var vslfound to N
  set var chk_hin to Y
  set var chk_name to N
  while chk_hin eq Y then
    newpage
    set var vvslhin text
    write "Enter incorrect vessel hin: " at 10 10
    write "(enter 'xxx' to search by vessel name)" at 12 5
    fillin vvslhin using " " at 10 37
    while vvslhin exists then
      if vvslhin eq xxx then
        set var chk_hin to N
        set var chk_name to Y
        break
      endif
      set pointer #1 e1 for vessel where vslhin = .vvslhin
      newpage
      if e1 eq 0 then
        *(vessel exists in database)
        set var vvslno to vslno in #1
        set var vvslname to vslname in #1
        set var vvslmake to vslmake in #1
        set var vvslmod to vslmodel in #1
        set var vvslyear to vslyear in #1
        set var vvslton to vsltons in #1
        set var vvslft to vsllenft in #1
        set var vvslin to vsllenin in #1
        set var vvslhp to vslhp in #1
        set var vvsluse to vsluse in #1
        set var vvsltype to vsltype in #1
        set var vvslprop to vslprop in #1
        set var vvslhull to vslhull in #1
        set var vvsleng to vsleng in #1
        set var vvslfuel to vslfuel in #1
        set var vconstr to vslconst in #1
        set var voffno to vsloffno in #1
        draw vhinform with all at 1
        write "Correct vessel? (Y)" at 22 20
        fillin vfound using "(" at 22 36
        if vfound fails or vfound eq Y then
          set var chk_hin to N
          set var vslfound to Y
          break
        else
          newpage
          write "Do you wish to reenter the HIN? (Y)" at 12 20
          fillin reenter using "(" at 12 52
          if reenter exists and reenter eq N then
            set var chk_hin to N
            set var chk_name to Y
            break
          endif
        endif
      else
        newpage
        write "This HIN does not exist in the database" at 10 20
        write "Do you wish to reenter the HIN? (Y)" at 12 20
        fillin correct using "(" at 12 52
        if correct exists and correct eq N then
          set var chk_hin to N
          set var chk_name to Y
        endif
        break
      endif
    endwhile
  endwhile
  while chk_name eq Y then
    newpage
    write "Enter name of vessel requiring HIN correction: " at 10 10
    fillin vvslname using " " at 10 56
    while vvslname exists then
      set pointer #1 e1 for vessel where vslname = .vvslname
      newpage
      while e1 eq 0 then
        *(vessel exists in database)
        set var vvslno to vslno in #1
      set var vvslname to vslname in #1
      set var vvslhin to vslhin in #1
      set var vvslmake to vslmake in #1
      set var vvslmod to vslmodel in #1
      set var vvslyear to vslyear in #1
      set var vvslton to vsltons in #1
      set var vvslft to vsllenft in #1
      set var vvslin to vsllenin in #1
      set var vvslhp to vslhp in #1
      set var vvsluse to vsluse in #1
      set var vvsltype to vsltype in #1
      set var vvslprop to vslprop in #1
      set var vvslhull to vslhull in #1
      set var vvsleng to vsleng in #1
      set var vvslfuel to vslfuel in #1
      set var vconstr to vslconst in #1
      set var voffno to vsloffno in #1
      newpage
      draw vhinform with all at 1
      write "Correct vessel? (N)" at 22 20
      fillin foundv using "(" at 22 36
      if foundv exists and foundv eq Y then
        set var chk_name to N
        set var vslfound to Y
        break
      endif
      next #1 e1
    endwhile
    if e1 ne 0 then
      newpage
      write "This vessel does not exist in the database" at 10 20
      write "Do you wish to reenter the vessel name? (Y)" at 12 20
      fillin correct using "(" at 12 60
      if correct exists and correct eq N then
        set var chk_name to N
      endif
    endif
    break
  endwhile
endwhile
if vslfound eq Y then
  set var vbadhin to .vvslhin
  set var error to Y
  while error eq Y then
    set var error to N
    newpage
    write "Press [PGDN] to change HIN or [ESC] to quit" at 1 1
    draw vhinform with all at 3
    edit var vvslhin return esc pgdn
    if #return eq pgdn then
      newpage
      change vslhin to .vvslhin in #1
      write "Return to correct HIN entered? (Y)" at 15 10
      fillin er using "(" at 15 41
      if er fails or er eq Y then
        set var error to Y
      else
        set pointer #1 e1 for owvsl where vhinown eq .vbadhin
        write "Changing values in owvsl"
        while e1 eq 0 then
          change vhinown to .vvslhin in #1
          next #1 e1
        endwhile
        set pointer #1 e1 for opvsl where vhinoper eq .vbadhin
        write "Changing values in opvsl"
        while e1 eq 0 then
          change vhinoper to .vvslhin in #1
          next #1 e1
        endwhile
        set pointer #1 e1 for sar where sarvhin eq .vbadhin
```

```
          write "Changing values in sar"
          while e1 eq 0 then
            change sarvhin to .vvslhin in #1
            next #1 e1
          endwhile
          set pointer #1 e1 for boarding where bovhin eq .vbadhin
          write "Changing values in boarding"
          while e1 eq 0 then
            change bovhin to .vvslhin in #1
            next #1 e1
          endwhile
        endif
      endif
    endwhile
  endif
  newpage
  write "More vessels to be changed? (N)" at 12 20
  fillin mrvsls using "(" at 12 48
  if mrvsls fails or mrvsls eq N then
    set var more to N
  endif
endwhile  *(end of main loop)
set error messages off
set null -0-
return
*( end )


$COMMAND
AM_VINT
*(*************************************************************************

    Program:               Add and Modify Vessel Intelligence
    Author:                H R Lang and J D Allen
    Date Written:          August 1987
    Description:           This program allows the user to input
                           and modify vessel intelligence
                           information.
    Tables Used:           vslintel
    Temp Tables:           none
    Reports Used:          none
    Forms Used:            vintform

*************************************************************************)

set error messages on
set messages off
set escape off
newpage
write "Program loading in progress.  Please stand by ......." at 10 10
clear all variables
set null ?
set var more to Y
while more eq Y then  *(main loop)
  newpage
  write "Do you wish to (A)dd or (M)odify information?" at 10 10
  fillin choice using ">" at 11 20
  if choice eq A then   *(add information)
    set var vintdate to .#date
    set var vepic text
    label redovsl
    newpage
    write "Press [PGDN] to load data or [ESC] to quit" at 1 1
    draw vintform with all at 3
    edit var return esc pgdn
    if #return eq pgdn then
      newpage
      write "Loading vslintel"
      set error messages off
      load vslintel
        .vvslname .vintdate .vepic .vstolen .vintrem1 .vintrem2 +
        .vintrem3
      end
      set error messages on
      write "Return to correct errors? (N)" at 15 10
      write "Note:  correct only errors indicated by messages" at 20 5
      fillin fixvsl using "(" at 15 36
      if fixvsl exists and fixvsl eq Y then
        goto redovsl
      endif
    endif
  endif
  while choice eq M then  *(modify information)
    newpage
    write "Enter name of vessel requiring changes then press [PGDN] +
to see if" at 1 1
    write "vessel exists in the database or [ESC] to quit" at 2 1
    draw vintform with all at 4
    edit var vvslname return esc pgdn
    if #return eq pgdn then
      set pointer #1 e1 for vslintel where intvname eq .vvslname
      while e1 eq 0 then
        set var vvslname to intvname in #1
        set var vintdate to intdate in #1
        set var vepic to epiccode in #1
        set var vstolen to stolen in #1
        set var vintrem1 to vremark1 in #1
        set var vintrem2 to vremark2 in #1
        set var vintrem3 to vremark3 in #1
        newpage
        draw vintform with all at 1
        write "Correct vessel? (Y)" at 23 25
        fillin correct using "(" at 23 41
        if correct fails or correct eq Y then
          label corvsl
          newpage
          write "Press [PGDN] to change database or [ESC] to quit" +
          at 1 1
          draw vintform with all at 3
          edit var return esc pgdn
          if #return eq pgdn then
            newpage
            write "changing values in vslintel"
            set error messages off
            change intvname to .vvslname in #1
            change intdate to .vintdate in #1
            change epiccode to .vepic in #1
            change stolen to .vstolen in #1
            change vremark1 to .vintrem1 in #1
            change vremark2 to .vintrem2 in #1
            change vremark3 to .vintrem3 in #1
            set error messages on
            write "Return to correct errors? (N)" at 15 10
            write "Note:  correct only errors indicated by messages" +
            at 20 5
            fillin vslfix using "(" at 15 36
            if vslfix exists and vslfix eq Y then
              goto corvsl
            endif
          endif
          set var choice to N
          break
        endif
        next #1 e1
      endwhile
      if e1 ne 0 then
        newpage
        write "This vessel does not exist in the vessel intelligence +
file" at 5 10
        write "Do you wish to reenter name? (Y)" at 15 10
        fillin reenter using "(" at 15 39
        if reenter exists and reenter eq N then
          break
        else
          clear vvslname
          clear vintdate
          clear vepic
          clear vstolen
          clear vintrem1
          clear vintrem2
          clear vintrem3
        endif
      endif
    else
      break
    endif
  endwhile
  newpage
  write "More intelligence to enter/modify? (N)" at 15 20
  fillin mrintel using "(" at 15 55
  if mrintel fails or mrintel eq Y then
    clear all variables
    set var more to Y
  else
    set var more to N
  endif
endwhile  *(end of main loop)
set error messages off
set null -0-
return
*( end )


$COMMAND
AM_PINT
*(**********************************************************************

    Program:          Add and Modify Personnel Intelligence
    Author:           H R Lang and J D Allen
    Date Written:     August 1987
    Description:      This program allows the user to input
                      and modify personnel intelligence
                      information.
    Tables Used:      perintel
    Temp Tables:      none
    Reports Used:     none
    Forms Used:       pintform

***********************************************************************)

set error messages on
set messages off
set escape off
newpage
write "Program loading in progress.  Please stand by ......." at 10 10
clear all variables
set null ?
set var more to Y
while more eq Y then  *(main loop)
  newpage
  write "Do you wish to (A)dd or (M)odify information?" at 10 10
  fillin choice using ">" at 11 20
  if choice eq A then  *(add information)
    newpage
    set var vintdate to .#date
    set var vintdob to 01/01/1901
    write "Press [PGDN] to load data or [ESC] to quit" at 1 1
    draw pintform with all at 3
    edit var return esc pgdn
    if #return eq pgdn then
      newpage
      write "Loading perintel"
      load perintel
        .vlname .vintdob .vintrem1 .vintrem2 .vintrem3 .vintdate +
        .vfname .vintmi
      end
    endif
  endif
  while choice eq M then  *(modify information)
    newpage
    write "Enter last and first names of individual requiring +
changes, then press" at 1 1
    write "[PGDN] to see if person exists in the database or [ESC] +
to quit" at 2 1
    draw pintform with all at 4
    edit var vlname vfname return esc pgdn
    if #return eq pgdn then
      set pointer #1 e1 for perintel where intlname eq .vlname and +
      intfname eq .vfname
      while e1 eq 0 then
        set var vlname to intlname in #1
        set var vfname to intfname in #1
        set var vintdob to intdob in #1
        set var vintmi to intmi in #1
        set var vintrem1 to pintrem1 in #1
        set var vintrem2 to pintrem2 in #1
        set var vintrem3 to pintrem3 in #1
        set var vintdate to pintdate in #1
        newpage
        draw pintform with all at 1
        write "Correct individual (Y)" at 23 25
        fillin correct using "(" at 23 44
        if correct fails or correct eq Y then
          newpage
          write "Press [PGDN] to change database or [ESC] to quit" +
          at 1 1
          draw pintform with all at 3
          edit var return esc pgdn
          if #return eq pgdn then
            set error messages off
            change intlname to .vlname in #1
            change intdob to .vintdob in #1
            change pintrem1 to .vintrem1 in #1
            change pintrem2 to .vintrem2 in #1
            change pintrem3 to .vintrem3 in #1
            change pintdate to .vintdate in #1
            change intfname to .vfname in #1
            change intmi to .vintmi in #1
            set error messages on
          endif
          set var choice to N
          break
        endif
        next #1 e1
      endwhile
      if e1 ne 0 then
        newpage
        write "This individual does not exist in the personnel +
intelligence file" at 5 10
        write "Do you wish to reenter name? (Y)" at 15 10
        fillin reenter using "(" at 15 39
        if reenter exists and reenter eq N then
          break
        else
          clear vlname
          clear vfname
          clear vintdob
          clear vintrem1
          clear vintrem2
          clear vintrem3
          clear vintmi
          clear vintdate
        endif
      endif
    else
      break
    endif
  endwhile
  newpage
  write "More intelligence to enter/modify? (Y)" at 15 20
  fillin mrintel using "(" at 15 55
  if mrintel fails or mrintel eq Y then
    clear all variables
    set var more to Y
  else
    set var more to N
  endif
endwhile  *(end of main loop)
set error messages off
set null -0-
return
*( end )
```

APPENDIX I
SREPORTS SOURCE CODE

```
$COMMAND
RPTMAIN
*(**********************************************************************

                     US COAST GUARD GROUP MONTEREY
         SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                     INTELLIGENCE SYSTEM (SARELTIS)

                 REPORTS MAIN MENU APPLICATION PROGRAM


                 Author:    Jon D. Allen, LT, USCG
                            Heidi R. Lang, LT, USN
                            AUGUST 1987

***********************************************************************)

set error messages off
set messages off
set escape off
clear all variables
newpage
set var pick4 integer
newpage
choose pick4 from rpt_menu in sreports.apx
if pick4 eq 0 then
   quit to restrpts.cmd
endif
if pick4 eq 1 then
   run INT_RPTS in sreports.apx
   quit to restrpts.cmd
   endif
if pick4 eq 2 then
   run BRD_RPTS in sreports.apx
   quit to restrpts.cmd
endif
if pick4 eq 3 then
   run SAR_RPTS in sreports.apx
   quit to restrpts.cmd
endif
if pick4 eq 4 then
   run VSL_RPTS in sreports.apx
   quit to restrpts.cmd
endif
if pick4 eq 5 then
   run PRS_RPTS in sreports.apx
   quit to restrpts.cmd
endif
if pick4 eq 6 then
   quit to restmain.cmd
endif
quit to restrpts.cmd
return
$menu
rpt_menu
column SARELTIS --- REPORTS MENU
INTELLIGENCE REPORTS
BOARDING REPORTS
SAR REPORTS
VESSEL REPORTS
PERSONNEL REPORTS
```

```
RETURN TO MAIN MENU
*( end )


$COMMAND
INT_RPTS
*(**********************************************************************

    Program:            Access Intelligence Reports
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This program calls the intelligence reports
                        master menu and application programs.
    Tables Used:        none
    Temp Tables:        none
    Reports Used:       none
    Forms Used:         none

***********************************************************************)

run INT_MAIN in intrpts.apx
return
*( end )


$COMMAND
BRD_RPTS
*(**********************************************************************

    Program:            Access Boarding Reports
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This program calls the boarding reports
                        master menu and application programs.
    Tables Used:        none
    Temp Tables:        none
    Reports Used:       none
    Forms Used:         none

***********************************************************************)

run BRD_MAIN in brdrpts.apx
return
*( end )


$COMMAND
SAR_RPTS
*(**********************************************************************

    Program:            Access Sar Reports
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This program calls the SAR reports master
                        menu and application programs.
    Tables Used:        none
    Temp Tables:        none
    Reports Used:       none
    Forms Used:         none

***********************************************************************)

run SAR_MAIN in sarrpts.apx
return
*( end )
```

```
$COMMAND
VSL_RPTS
*(**********************************************************************

    Program:           Access Vessel Reports
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program calls the vessel reports
                       master menu and application programs.
    Tables Used:       none
    Temp Tables:       none
    Reports Used:      none
    Forms Used:        none

***********************************************************************)

run VSL_MAIN in vslrpts.apx
return
*( end )


$COMMAND
PRS_RPTS
*(**********************************************************************

    Program:           Access Personnel Reports
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program calls the personnel reports
                       master menu and application programs.
    Tables Used:       none
    Temp Tables:       none
    Reports Used:      none
    Forms Used:        none

***********************************************************************)

run PRS_MAIN in prsrpts.apx
return
*( end )
```

APPENDIX J
INTRPTS SOURCE CODE

```
$COMMAND
INT_MAIN
*(**********************************************************************

                        US COAST GUARD GROUP MONTEREY
            SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                        INTELLIGENCE SYSTEM (SARELTIS)

             INTELLIGENCE REPORTS MENU and APPLICATION PROGRAMS

                    Author:    Jon D. Allen, LT, USCG
                               Heidi R. Lang, LT, USN
                               AUGUST 1987

**********************************************************************)

set error messages off
set messages off
set escape off
clear all variables
newpage
choose pick5 from int_menu in intrpts.apx
if pick5 eq 0 then
   quit to restint.cmd
endif
if pick5 eq 1 then
   run EPICDUMP in intrpts.apx
   quit to restint.cmd
endif
if pick5 eq 2 then
   run INTELRPT in intrpts.apx
   quit to restint.cmd
endif
if pick5 eq 3 then
   run PERINTEL in intrpts.apx
   quit to restint.cmd
endif
if pick5 eq 4 then
   quit to restrpts.cmd
endif
if pick5 eq 5 then
   quit to restint.cmd
endif
quit to restint.cmd
return
$menu
int_menu
column SARELTIS --- INTELLIGENCE REPORTS
VESSEL INTELLIGENCE ANALYSIS (EPIC)
VESSEL PROFILE FOR INDIVIDUAL VIOLATIONS ANALYSIS
PERSONNEL INTELLIGENCE SUMMARY
RETURN TO REPORTS MENU
*( end )


$COMMAND
PERINTEL
*(**********************************************************************
    Program:            Personnel Intelligence Dump Program
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This program performs a listing by date and
                        by alphabtic search of those personnel in
                        the table PERINTEL (personnel intelligence).
    Tables Used:        perintel
    Temp Tables:        persum
    Reports Used:       perintel
    Forms Used:         none
***********************************************************************)

set null " "
set error messages off
set messages off
set escape on
newpage
clear all variables
set error messages off
write "This program will perform dump intelligence information" at 3 1
write "on people that are in the PERINTEL table.  Information" at 4 1
write "may be selected by alphabetic and intelligence date." at 5 1
set var strtdate date
set var enddate date
set var datediff to 0
set error messages on
while datediff <= 0 then
   write "Please enter starting date in the form MM/DD/YYYY:" at 8 1
   fillin strtdate using " " at 8 53
   write "Please enter ending date in the form MM/DD/YYYY:" at 10 1
   fillin enddate using " " at 10 53
   set var datediff to .enddate - .strtdate
endwhile
newpage
write "You will now be asked to provide beginning and ending " at 3 1
write "letters to start and end the alphabetic search.  If you " at 4 1
write "enter an [a] and a [z] you will get all of the personnel" at 5 1
write "in the the table PERINTEL whose intelligence date is " at 6 1
write "between the dates that you previously entered." at 7 1
set var strtltr text
set var endltr text
set var ltrdiff to 0
while ltrdiff = 0 then
   fillin strtltr using "Please enter starting letter(s):" at 10 1
   fillin endltr using "Please enter ending letter(s):" at 12 1
   if endltr lt .strtltr then
      set var ltrdiff to 0
   else
      break
   endif
endwhile
set error messages off
remove persum
newpage
clear numrows
set var numrows integer
project persum from perintel using all sorted by intlname +
where pintdate ge .strtdate and pintdate le .enddate and +
intlname ge .strtltr and intlname le .endltr
compute numrows as rows from persum
newpage
write "Now preparing report"
if numrows fails or numrows = 0 then
   write "No records exist that match your selection criteria" at 10 10
   write "Please press any key to continue" at 16 15
   pause
else
   output printer with perintel.rpt
   newpage
   print perintel
   output screen
endif
remove persum
newpage
clear all variables
return
*( end )


$COMMAND
EPICDUMP
*(**********************************************************************

    Program:            EPIC Intelligence Analysis
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        Performs a dump of all vessels in the
                        VSLINTEL table within a specified time
                        period.  Four reports are generated
                        providing listings by EPIC Code, by
                        stolen vessels, and by intelligence
                        remarks, and a frequency breakdown
                        and analysis of EPIC Codes.
    Tables Used:        vslintel
    Temp Tables:        epicsum, stolesum, intelsum, epictmp1,
                        epictmp2, epictmp
    Reports Used:       epicsum, stolesum, intelsum, epicrpt
    Forms Used:         none

**********************************************************************)

set error messages off
set messages off
set escape on
newpage
clear all variables
set null " "
newpage
write "This program performs a dump of all vessels,  by EPIC +
Code/Stolen" at 2 1
write "codes,  which intelligence information has been collected." +
at 3 1
write "Information to be analyzed can be selected within a specified" +
at 4 1
write "time period." at 5 1
set var strtdate date
set var enddate date
set var datediff to 0
set error messages on
while datediff <= 0 then
   write "Please enter starting date in the form MM/DD/YYYY:" at 8 1
   fillin strtdate using " " at 8 51
   write "Please enter ending date in the form MM/DD/YYYY:" at 10 1
   fillin enddate using " " at 10 51
   set var datediff to .enddate - .strtdate
endwhile
set error messages off
remove epicsum
remove stolesum
remove intelsum
newpage
project epicsum from vslintel using epiccode intdate intvname stolen +
sorted by epiccode intvname where intdate >= .strtdate and +
intdate <= .enddate and epiccode exists
compute numrows as rows from epicsum
newpage
write "Now preparing sorted vessel EPIC Code report......" at 10 1
if numrows fails or numrows = 0 then
```

```
   write "No records exist that match your selection criteria" at 10 10
   write "Please press any key to continue" at 16 15
   pause
else
   output printer with epicsum.rpt
   newpage
   print epicsum
   output screen
endif
remove epicsum
newpage
clear numrows
project stolesum from vslintel using intdate intvname stolen sorted by +
intvname where intdate >= .strtdate and intdate <= .enddate and +
stolen = Y
compute numrows as rows from stolesum
write "Now preparing sorted stolen vessel report......" at 10 1
if numrows fails or numrows = 0 then
   write "No records exist that match your selection criteria" at 10 10
   write "Please press any key to continue" at 16 15
   pause
else
   output printer with stolesum.rpt
   newpage
   print stolesum
   output screen
endif
remove stolesum
newpage
clear numrows
project intelsum from vslintel using intdate vremark1 vremark2 +
vremark3 intvname sorted by intvname intdate where vremark1 exists +
or vremark2 exists or vremark3 exists
compute numrows as rows from intelsum
write "Now preparing sorted vessel intelligence remarks report" at 10 1
if numrows fails or numrows = 0 then
   write "No records exist that match your selection criteria" at 10 10
   write "Please press any key to continue" at 16 15
   pause
else
   output printer with intelsum.rpt
   newpage
   print intelsum
   output screen
endif
remove intelsum
remove epictmp1
remove epictmp2
newpage
project epictmp1 from vslintel using epiccode intdate where intdate +
>= .strtdate and intdate <= .enddate and epiccode exists
project epictmp2 from epic using all sorted by epicno
remove epictmp
define
attribute
ecpct real
ecnumber integer
relation
epictmp with epicno epicdsc1 epicdsc2 epicdsc3 epicdsc4 ecnumber ecpct
end
compute totepics as count epiccode from epictmp1 where epiccode exists
set var pcttmp1 real
set pointer #1 n1 for epictmp2
while n1 eq 0 then
   set var epictest to epicno in #1
   set var edsc1 to epicdsc1 in #1
   set var edsc2 to epicdsc2 in #1
   set var edsc3 to epicdsc3 in #1
   set var edsc4 to epicdsc4 in #1
   compute epicnum as count epiccode from epictmp1 where +
epiccode = .epictest
   if totepics = 0 then
      set var pcttmp1 to .epicnum / 1.0     *(prevents division by 0)
   else
      set var pcttmp1 to .epicnum / .totepics
   endif
   set var pcttmp2 to .pcttmp1 x 100.0
   load epictmp
      .epictest .edsc1 .edsc2 .edsc3 .edsc4 .epicnum .pcttmp2
   end
   next #1 n1
endwhile
write "Now preparing vessel EPIC code frequency report......" at 10 1
output printer with epicrpt.rpt
newpage
print epicrpt
newpage
output screen
remove epictmp2
remove epictmp
remove epictmp1
newpage
clear all variables
return
*( end )


$COMMAND
INTELRPT
*(**********************************************************************

    Program:            Violation Vessel Profile Intelligence
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This program examines vessel boardings to
                        create a profile of vessel which violate
                        specific laws during a specified time period.
    Tables Used:        boarding, vessel, viols, boattype, boatuse,
                        constr, engcomp, fuelcomp, hullmat, boatprop
    Temp Tables:        bosum, vslsum, violsum, vslmo,
                        summary1, summary2, summary3, summary4,
                        summary5, summary6, summary7, temp
    Reports Used:       vtypemo, vusemo, vconstmo, vengmo,
                        vfuelmo, vhullmo, vpropmo, vlenftmo,
                        vyearmo
    Forms Used:         none

**********************************************************************)

set null " "
set error messages off
set messages off
set escape on
newpage
clear all variables
remove vslsum
remove bosum
remove violsum
remove temp
remove summary1
remove summary2
remove summary3
remove summary4
remove summary5
remove summary6
remove summary7
newpage
clear all variables
write "This module performs a statistical analysis of vessel " at 2 1
write "intelligence information during a specified time period " at 3 1
set var strtdate date
set var enddate date
set var datediff to 0
set error messages on
while datediff <= 0 then
   write "Please enter starting date in the form MM/DD/YYYY:" at 8 1
   fillin strtdate using " " at 8 51
   write "Please enter ending date in the form MM/DD/YYYY:" at 10 1
   fillin enddate using " " at 10 51
   set var datediff to .enddate - .strtdate
endwhile
set error messages off
write "Now finding information which meets date criteria......" at 14 1
project bosum from boarding using bodate bono viol54 viol55 viol56 +
viol57 viol58 viol59 viol60 viol61 viol62 viol63 viol64 viol65 viol66 +
viol67 viol68 uns69 uns70 uns71 uns72 uns73 uns74 uns75 uns76 uns77 +
uns78 uns79 bovhin weapons where bodate >= .strtdate and +
bodate <= .enddate
project vslsum from vessel using vslconst vsleng vslfuel vslhull +
vsllenft vslprop vsltype vsluse vslyear vsltons vslhin
join bosum using bovhin with vslsum using vslhin forming violsum +
where eq
remove bosum
remove vslsum
compute totbrds as count bono from violsum where bono exists
compute numrows as rows from violsum
newpage
set var code to 0
set error messages on
while code < 54 or code > 80 then
   write "Please enter in violation code or unsafe condition that " +
at 1 1
   write "you wish to run a vessel desription statistical summary:" +
at 2 1
   write "54 - Numbering                    69 - Unsafe Conditions +
Causing" at 4 1
   write "55 - Certificate                       Termination" at 5 1
   write "56 - Personal Flotation Device    70 - Corrected on the +
spot" at 6 1
   write "57 - Sound Producing Device       71 - Overloaded" at 7 1
   write "58 - Bell                         72 - Accumulation of fuel +
in bilges" at 8 1
   write "59 - Fire Extinguisher            73 - Fuel Leak" at 9 1
   write "60 - Flame Control                74 - Manifestly Unsafe +
Voyage" at 10 1
   write "61 - Ventilation                  75 - Hazardous Bars +
(13th District)" at 11 1
   write "62 - Nav./Anchor Lights           76 - Alcohol Condition" +
at 12 1
   write "63 - Negligent Operation          77 - Other" at 13 1
   write "64 - Failure to Terminate         78 - Warning Issued" at 14 1
   write "65 - Pollution Placard Not Posted 79 - No Violation" at 15 1
   write "66 - Marine Sanitation Devices    80 - Weapons On Board" +
at 16 1
   write "67 - Visual Distress Signals" at 17 1
   write "68 - Document/Official Numbers" at 18 1
   fillin code using "Please enter code: " at 20 24
endwhile
set error messages off
remove vslmo
newpage
write "Now evaluating vessel profile.........." at 11 1
        *(this next section of code finds the information only
          dealing with the specific violation code)
if code = 54 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where viol54 exists
endif
if code = 55 then
```

```
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where viol55 exists
endif
if code = 56 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where viol56 exists
endif
if code = 57 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where viol57 exists
endif
if code = 58 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where viol58 exists
endif
if code = 59 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where viol59 exists
endif
if code = 60 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where viol60 exists
endif
if code = 61 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where viol61 exists
endif
if code = 62 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where viol62 exists
endif
if code = 63 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where viol63 exists
endif
if code = 64 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where viol64 exists
endif
if code = 65 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where viol65 exists
endif
if code = 66 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where viol66 exists
endif
if code = 67 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where viol67 exists
endif
if code = 68 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where viol68 exists
endif
if code = 69 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where uns69 exists
endif
if code = 70 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where uns70 exists
endif
if code = 71 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where uns71 exists
endif
if code = 72 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where uns72 exists
```

```
endif
if code = 73 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where uns73 exists
endif
if code = 74 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where uns74 exists
endif
if code = 75 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where uns75 exists
endif
if code = 76 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where uns76 exists
endif
if code = 77 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where uns77 exists
endif
if code = 78 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where uns78 exists
endif
if code = 79 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where uns79 exists
endif
if code = 80 then
   project vslmo from violsum using vslconst vsleng vslfuel vslhull +
   vsllenft vslprop vsltype vsluse vslyear bono where weapons exists
endif
compute numrows as rows from vslmo
remove violsum
remove summary1
remove summary2
remove summary3
remove summary4
newpage
remove summary5
remove summary6
remove summary7
newpage
if numrows fails or numrows = 0 then
   write "No records exist that match your selection criteria" at 10 10
   write "Please press any key to continue" at 16 15
   pause
else
        *(Now getting violation code description)
   set pointer #1 n1 for viols
   while n1 eq 0 then
      set var vno to violno in #1
      if code = .vno then
         set var vltndesc to vioIdesc in #1
         break
      endif
      next #1 n1
   endwhile
   newpage
   write "Preparing vessel type demographics.................."
   define
   attributes
   counter integer
   typedesc text 15
   relation
   summary1 with vsltype typedesc counter
   end
   set pointer #1 n1 for boattype
   while n1 eq 0 then
   set var typetest to vtype in #1
   set var typdesc1 to vtypedes in #1
   compute sumtype as count vsltype from vslmo where +
   vsltype = .typetest
   if sumtype fails then
      set var sumtype to 0
   endif
   load summary1
      .typetest .typdesc1 .sumtype
   end
   next #1 n1
endwhile
compute totviols as sum counter from summary1
output printer with vtypemo.rpt
newpage
print vtypemo
output screen
remove summary1
write "Preparing vessel use demographics..................."
define
attributes
numuse integer
usedesc text 15
relations
summary2 with vsluse usedesc numuse
end
set pointer #1 n1 for boatuse
while n1 eq 0 then
   set var usetest to vuse in #1
   set var usedesc1 to vusedesc in #1
   compute sumuse as count vsluse in vslmo where vsluse = .usetest
   if sumuse fails then
      set var sumuse to 0
   endif
   load summary2
      .usetest .usedesc1 .sumuse
   end
   next #1 n1
endwhile
compute totviols as sum numuse from summary2
output printer with vusemo.rpt
newpage
print vusemo
output screen
remove summary2
write "Preparing vessel construction type demographics......"
define
attributes
numconst integer
constdes text 10
relations
summary3 with vslconst constdes numconst
end
set pointer #1 n1 for constr
while n1 eq 0 then
   set var consttst to constype in #1
   set var constrds to consdesc in #1
   compute sumconst as count vslconst in vslmo where +
   vslconst = .consttst
   if sumconst fails then
      set var sumconst to 0
   endif
   load summary3
      .consttst .constrds .sumconst
   end
   next #1 n1
endwhile
compute totviols as sum numconst from summary3
output printer with vconstmo.rpt
newpage
```

```
print vconstmo
output screen
remove summary3
write "Preparing vessel engine compartment demographics...."
define
attributes
numeng integer
engdesc text 10
relations
summary4 with vsleng engdesc numeng
end
set pointer #1 n1 for engcomp
while n1 eq 0 then
   set var engtst to ecomptyp in #1
   set var engdescr to ecompdes in #1
   compute sumeng as count vsleng in vslmo where vsleng = .engtst
   if sumeng fails then
      set var sumeng to 0
   endif
   load summary4
      .engtst .engdescr .sumeng
   end
   next #1 n1
endwhile
compute totviols as sum numeng from summary4
output printer with vengmo.rpt
newpage
print vengmo
output screen
remove summary4
write "Preparing fuel compartment type demographics........."
define
attributes
numfuel integer
fueldesc text 10
relations
summary5 with vslfuel fueldesc numfuel
end
set pointer #1 n1 for fuelcomp
while n1 eq 0 then
   set var fueltst to fcomptyp in #1
   set var fdesc to fcompdes in #1
   compute sumfuel as count vslfuel in vslmo where vslfuel = .fueltst
   if sumfuel fails then
      set var sumfuel to 0
   endif
   load summary5
      .fueltst .fdesc .sumfuel
   end
   next #1 n1
endwhile
compute totviols as sum numfuel from summary5
output printer with vfuelmo.rpt
newpage
print vfuelmo
output screen
remove summary5
write "Preparing hull material demographics..............."
define
attributes
numhull integer
hulldesc text 15
relations
summary6 with vslhull hulldesc numhull
end
set pointer #1 n1 for hullmat
while n1 eq 0 then
   set var hulltst to hmattype in #1
   set var hdesc to hmatdesc in #1
   compute sumhull as count vslhull in vslmo where vslhull = .hulltst
```

```
      if sumhull fails then
         set var sumhull to 0
      endif
      load summary6
         .hulltst .hdesc .sumhull
      end
      next #1 n1
   endwhile
   compute totviols as sum sumhull from summary6
   output printer with vhullmo.rpt
   newpage
   print vhullmo
   output screen
   remove summary6
   write "Preparing vessel propulsion demographics..........."
   define
   attributes
   numprop integer
   descprop text 15
   relations
   summary7 with vslprop descprop numprop
   end
   set pointer #1 n1 for boatprop
   while n1 eq 0 then
      set var proptst to proptype in #1
      set var prdesc to propdesc in #1
      compute sumprop as count vslprop in vslmo where vslprop = .proptst
      if sumprop fails then
         set var sumprop to 0
      endif
      load summary7
         .proptst .prdesc .sumprop
      end
      next #1 n1
   endwhile
   compute totviols as sum numprop from summary7
   output printer with vpropmo.rpt
   newpage
   print vpropmo
   output screen
   remove summary7
   write "Preparing vessel length demographics..............."
   compute totvsls as count bono from vslmo where vsllenft exists
   compute len<20 as count bono from vslmo where vsllenft < 20
   compute len2030 as count bono from vslmo where vsllenft >= 20 and +
vsllenft < 30
   compute len3040 as count bono from vslmo where vsllenft >= 30 and +
vsllenft < 40
   compute len4050 as count bono from vslmo where vsllenft >= 40 and +
vsllenft < 50
   compute len5060 as count bono from vslmo where vsllenft >= 50 and +
vsllenft < 60
   compute len6070 as count bono from vslmo where vsllenft >= 60 and +
vsllenft < 70
   compute len7080 as count bono from vslmo where vsllenft >= 70 and +
vsllenft < 80
   compute len8090 as count bono from vslmo where vsllenft >= 80 and +
vsllenft < 90
   compute len90100 as count bono from vslmo where vsllenft >= 90 and +
vsllenft < 100
   compute len>100 as count bono from vslmo where vsllenft >= 100
*(computing total number of violations)
   set var tv1 to .len<20 + .len2030
   set var tv2 to .len3040 + .len4050
   set var tv3 to .len5060 + .len6070
   set var tv4 to .len7080 + .len8090
   set var tv5 to .len>100 + .len90100
   set var tv10 to .tv1 + .tv2
   set var tv20 to .tv3 + .tv4
   set var tv30 to .tv5 + .tv10
   set var totviols to .tv20 + .tv30
   set var pct<20 real
   set var pct2030 real
   set var pct3040 real
   set var pct4050 real
   set var pct5060 real
   set var pct6070 real
   set var pct7080 real
   set var pct8090 real
   set var pct90100 real
   set var pct>100 real
*(Evaluating percentages based on total boardings)
   set var pct<20 to .len<20 / .totbrds
   set var pct2030 to .len2030 / .totbrds
   set var pct3040 to .len3040 / .totbrds
   set var pct4050 to .len4050 / .totbrds
   set var pct5060 to .len5060 / .totbrds
   set var pct6070 to .len6070 / .totbrds
   set var pct7080 to .len7080 / .totbrds
   set var pct8090 to .len8090 / .totbrds
   set var pct90100 to .len90100 / .totbrds
   set var pct>100 to .len>100 / .totbrds
   set var vpct<20 real
   set var vpct2030 real
   set var vpct3040 real
   set var vpct4050 real
   set var vpct5060 real
   set var vpct6070 real
   set var vpct7080 real
   set var vpct8090 real
   set var vpct9010 real
   set var vpct>100 real
   *(Evaluating percentages based on total violations)
   set var vpct<20 to .len<20 / .totviols
   set var vpct2030 to .len2030 / .totviols
   set var vpct3040 to .len3040 / .totviols
   set var vpct4050 to .len4050 / .totviols
   set var vpct5060 to .len5060 / .totviols
   set var vpct6070 to .len6070 / .totviols
   set var vpct7080 to .len7080 / .totviols
   set var vpct8090 to .len8090 / .totviols
   set var vpct9010 to .len90100 / .totviols
   set var vpct>100 to .len>100 / .totviols
   set var pcttbrds real
   set var pcttbrds to .totviols / .totbrds
*(Defining temporary table for printing)
   remove temp
   define
   attributes
   tmp text 2
   relations
   temp with tmp
   end
   set var tmp to XX
   load temp
      .tmp
   end
   output printer with vlenftmo.rpt
   newpage
   print vlenftmo
   output screen
   write "Preparing vessel year demographics................."
   compute yr<1960 as count bono in vslmo where vslyear < 1960
   compute yr<1965 as count bono in vslmo where vslyear >= 1960 and +
   vslyear < 1965
   compute yr<1970 as count bono in vslmo where vslyear >= 1965 and +
   vslyear < 1970
   compute yr<1975 as count bono in vslmo where vslyear >= 1970 and +
   vslyear < 1975
   compute yr<1980 as count bono in vslmo where vslyear >= 1975 and +
   vslyear < 1980
   compute yr<1985 as count bono in vslmo where vslyear >= 1980 and +
   vslyear < 1985
   compute yr<1990 as count bono in vslmo where vslyear >= 1985 and +
   vslyear < 1990
   compute yr<1995 as count bono in vslmo where vslyear >= 1990 and +
   vslyear < 1995
   compute yr<2000 as count bono in vslmo where vslyear >= 1995 and +
   vslyear < 2000
   compute yr>2000 as count bono in vslmo where vslyear >= 2000
   set var vpct<60 real
   set var vpct<65 real
   set var vpct<70 real
   set var vpct<75 real
   set var vpct<80 real
   set var vpct<85 real
   set var vpct<90 real
   set var vpct<95 real
   set var vpct<00 real
   set var vpct>00 real
   *(Evaluating percentages based on total violations)
   set var vpct<60 to .yr<1960 / .totviols
   set var vpct<65 to .yr<1965 / .totviols
   set var vpct<70 to .yr<1970 / .totviols
   set var vpct<75 to .yr<1975 / .totviols
   set var vpct<80 to .yr<1980 / .totviols
   set var vpct<85 to .yr<1985 / .totviols
   set var vpct<90 to .yr<1990 / .totviols
   set var vpct<95 to .yr<1995 / .totviols
   set var vpct<00 to .yr<2000 / .totviols
   set var vpct>00 to .yr>2000 / .totviols
   set var pct<60 real
   set var pct<65 real
   set var pct<70 real
   set var pct<75 real
   set var pct<80 real
   set var pct<85 real
   set var pct<90 real
   set var pct<95 real
   set var pct<00 real
   set var pct>00 real
   *(Evaluating percentages based on total boardings)
   set var pct<60 to .yr<1960 / .totbrds
   set var pct<65 to .yr<1965 / .totbrds
   set var pct<70 to .yr<1970 / .totbrds
   set var pct<75 to .yr<1975 / .totbrds
   set var pct<80 to .yr<1980 / .totbrds
   set var pct<85 to .yr<1985 / .totbrds
   set var pct<90 to .yr<1990 / .totbrds
   set var pct<95 to .yr<1995 / .totbrds
   set var pct<00 to .yr<2000 / .totbrds
   set var pct>00 to .yr>2000 / .totbrds
   output printer with vyearmo.rpt
   newpage
   print vyearmo
   newpage
   output screen
   remove temp
   remove vslmo
   remove summary1
   remove summary2
   remove summary3
   newpage
   remove summary4
   remove summary5
   remove summary6
   remove summary7
   newpage
   clear all variables
endif
```

```
return
*( end )
```

APPENDIX K
BRDRPTS SOURCE CODE

```
$COMMAND
BRD_MAIN
*(***********************************************************************

                      US COAST GUARD GROUP MONTEREY
         SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                      INTELLIGENCE SYSTEM (SARELTIS)

              BOARDING REPORTS MENU and APPLICATION PROGRAMS

                 Author:    Jon D. Allen, LT, USCG
                            Heidi R. Lang, LT, USN
                            AUGUST 1987

***********************************************************************)

set error messages off
set escape off
set messages off
clear all variables
newpage
set var pick6 integer
choose pick6 from brd_menu in brdrpts.apx
if pick6 eq 0 then
   quit to restbrd.cmd
endif
if pick6 eq 1 then
   run VIOLFREQ in brdrpts.apx
   quit to restbrd.cmd
endif
if pick6 eq 2 then
   run BOARDING in brdrpts.apx
   quit to restbrd.cmd
endif
if pick6 eq 3 then
   run CMERPT in brdrpts.apx
   quit to restbrd.cmd
endif
if pick6 eq 4 then
   run BOMAJOR in brdrpts.apx
   quit to restbrd.cmd
endif
if pick6 eq 5 then
   run BOMINOR in brdrpts.apx
   quit to restbrd.cmd
endif
if pick6 eq 6 then
   run SMBOATRP in brdrpts.apx
   quit to restbrd.cmd
endif
if pick6 eq 7 then
   run DAYNIGHT in brdrpts.apx
   quit to restbrd.cmd
endif
if pick6 eq 8 then
   quit to restrpts.cmd
endif
quit to restbrd.cmd
return
```

```
$menu
brd_menu
column SARELTIS --- BOARDING REPORTS
VIOLATION/UNSAFE CONDITION FREQUENCY
BOARDINGS SUMMARY LISTING
CME DECALS: ANALYSIS OF VESSELS BOARDED
LARGE SCALE AREA BOARDING LOCATION ANALYSIS
SMALL SCALE AND COASTLINE AREA BOARDING LOCATION ANALYSIS
SMALL BOAT OPERATIONS AREA BOARDING LOCATION ANALYSIS
TIME OF DAY BOARDING CASE ANALYSIS
RETURN TO REPORTS MENU
*( end )


$COMMAND
VIOLFREQ
*(**********************************************************************

    Program:           Violation / Unsafe Condition Frequency Analysis
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program performs a statistical analysis
                       of boarding violations and unsafe conditions
                       within a specified time period.
    Tables Used:       boarding
    Temp Tables:       frqchart, temp
    Reports Used:      violfreq, unsfreq
    Forms Used:        none

***********************************************************************)

set null " "
set error messages off
set escape on
set messages off
newpage
clear all variables
write "This module performs a statistical analysis of boarding " at 2 1
write "violations and unsafe conditions within a specified " at 3 1
write "time period." at 4 1
set var strtdate date
set var enddate date
set error messages on
set var datediff to 0
while datediff <= 0 then
   write "Please enter starting date in the form MM/DD/YYYY:" at 6 1
   fillin strtdate using " " at 6 53
   write "Please enter ending date in the form MM/DD/YYYY:" at 8 1
   fillin enddate using " " at 8 53
   set var datediff to .enddate - .strtdate
endwhile
set error messages off
project frqchart from boarding using bono viol54 viol55 viol56 viol57 +
   viol58 viol59 viol60 viol61 viol62 viol63 viol64 viol65 viol66 +
   viol67 viol68 uns69 uns70 uns71 uns72 uns73 uns74 uns75 uns76 +
   uns77 uns78 uns79 where bodate ge .strtdate and bodate le .enddate
compute viol54s as count viol54 from frqchart where viol54 exists
compute viol55s as count viol55 from frqchart where viol55 exists
compute viol56s as count viol56 from frqchart where viol56 exists
compute viol57s as count viol57 from frqchart where viol57 exists
compute viol58s as count viol58 from frqchart where viol58 exists
compute viol59s as count viol59 from frqchart where viol59 exists
compute viol60s as count viol60 from frqchart where viol60 exists
compute viol61s as count viol61 from frqchart where viol61 exists
compute viol62s as count viol62 from frqchart where viol62 exists
compute viol63s as count viol63 from frqchart where viol63 exists
compute viol64s as count viol64 from frqchart where viol64 exists
compute viol65s as count viol65 from frqchart where viol65 exists
compute viol66s as count viol66 from frqchart where viol66 exists
compute viol67s as count viol67 from frqchart where viol67 exists
compute viol68s as count viol68 from frqchart where viol68 exists
      *(computing total violations)
set var vtot1 to .viol54s + .viol55s
set var vtot2 to .viol56s + .viol57s
set var vtot3 to .viol58s + .viol59s
set var vtot4 to .viol60s + .viol61s
set var vtot5 to .viol62s + .viol63s
set var vtot6 to .viol64s + .viol65s
set var vtot7 to .viol66s + .viol67s
set var vtot8 to .vtot1 + .viol68s
set var vtot9 to .vtot2 + .vtot3
set var vtot10 to .vtot4 + .vtot5
set var vtot11 to .vtot6 + .vtot7
set var vtot12 to .vtot8 + .vtot9
set var vtot13 to .vtot10 + .vtot11
set var vtotal to .vtot12 + .vtot13
compute uns69s as count uns69 from frqchart where uns69 exists
compute uns70s as count uns70 from frqchart where uns70 exists
compute uns71s as count uns71 from frqchart where uns71 exists
compute uns72s as count uns72 from frqchart where uns72 exists
compute uns73s as count uns73 from frqchart where uns73 exists
compute uns74s as count uns74 from frqchart where uns74 exists
compute uns75s as count uns75 from frqchart where uns75 exists
compute uns76s as count uns76 from frqchart where uns76 exists
compute uns77s as count uns77 from frqchart where uns77 exists
compute uns78s as count uns78 from frqchart where uns78 exists
compute uns79s as count uns79 from frqchart where uns79 exists
     *(computing total unsafe conditions)
set var unstot1 to .uns69s + .uns70s
set var unstot2 to .uns71s + .uns72s
set var unstot3 to .uns73s + .uns74s
set var unstot4 to .uns75s + .uns76s
set var unstot5 to .uns77s + .unstot1
set var unstot6 to .unstot2 + .unstot3
set var unstot7 to .unstot4 + .unstot5
set var unstotal to .unstot6 + .unstot7
compute totbrds as count bono from frqchart where bono exists
set var viol54s real
set var viol55s real
set var viol56s real
set var viol57s real
set var viol58s real
set var viol59s real
set var viol60s real
set var viol61s real
set var viol62s real
set var viol63s real
set var viol64s real
set var viol65s real
set var viol66s real
set var viol67s real
set var viol68s real
set var uns69s real
set var uns70s real
set var uns71s real
set var uns72s real
set var uns73s real
set var uns74s real
set var uns75s real
set var uns76s real
set var uns77s real
set var uns78s real
set var uns79s real
set var totbrds real
     *(computing percentages based on total boardings)
set var pct54tmp to .viol54s / .totbrds
set var pct54 to .pct54tmp x 100.0
set var pct55tmp to .viol55s / .totbrds
set var pct55 to .pct55tmp x 100.0
set var pct56tmp to .viol56s / .totbrds
set var pct56 to .pct56tmp x 100.0
set var pct57tmp to .viol57s / .totbrds
set var pct57 to .pct57tmp x 100.0
set var pct58tmp to .viol58s / .totbrds
set var pct58 to .pct58tmp x 100.0
set var pct59tmp to .viol59s / .totbrds
set var pct59 to .pct59tmp x 100.0
set var pct60tmp to .viol60s / .totbrds
set var pct60 to .pct60tmp x 100.0
set var pct61tmp to .viol61s / .totbrds
set var pct61 to .pct61tmp x 100.0
set var pct62tmp to .viol62s / .totbrds
set var pct62 to .pct62tmp x 100.0
set var pct63tmp to .viol63s / .totbrds
set var pct63 to .pct63tmp x 100.0
set var pct64tmp to .viol64s / .totbrds
set var pct64 to .pct64tmp x 100.0
set var pct65tmp to .viol65s / .totbrds
set var pct65 to .pct65tmp x 100.0
set var pct66tmp to .viol66s / .totbrds
set var pct66 to .pct66tmp x 100.0
set var pct67tmp to .viol67s / .totbrds
set var pct67 to .pct67tmp x 100.0
set var pct68tmp to .viol68s / .totbrds
set var pct68 to .pct68tmp x 100.0
set var pct69tmp to .uns69s / .totbrds
set var pct69 to .pct69tmp x 100.0
set var pct70tmp to .uns70s / .totbrds
set var pct70 to .pct70tmp x 100.0
set var pct71tmp to .uns71s / .totbrds
set var pct71 to .pct71tmp x 100.0
set var pct72tmp to .uns72s / .totbrds
set var pct72 to .pct72tmp x 100.0
set var pct73tmp to .uns73s / .totbrds
set var pct73 to .pct73tmp x 100.0
set var pct74tmp to .uns74s / .totbrds
set var pct74 to .pct74tmp x 100.0
set var pct75tmp to .uns75s / .totbrds
set var pct75 to .pct75tmp x 100.0
set var pct76tmp to .uns76s / .totbrds
set var pct76 to .pct76tmp x 100.0
set var pct77tmp to .uns77s / .totbrds
set var pct77 to .pct77tmp x 100.0
set var pct78tmp to .uns78s / .totbrds
set var pct78 to .pct78tmp x 100.0
set var pct79tmp to .uns79s / .totbrds
set var pct79 to .pct79tmp x 100.0
set var viol54s integer
set var viol55s integer
set var viol56s integer
set var viol57s integer
set var viol58s integer
set var viol59s integer
set var viol60s integer
set var viol61s integer
set var viol62s integer
set var viol63s integer
set var viol64s integer
set var viol65s integer
set var viol66s integer
set var viol67s integer
set var viol68s integer
set var uns69s integer
set var uns70s integer
set var uns71s integer
set var uns72s integer
set var uns73s integer
set var uns74s integer
set var uns75s integer
set var uns76s integer
set var uns77s integer
set var uns78s integer
set var uns79s integer
set var totbrds integer
remove temp
     *(creating temporary table for printing purposes)
newpage
define
attributes
pgcount integer
relations
temp with pgcount
end
load temp
"1"
end
newpage
write "Preparing reports........"
output printer with violfreq.rpt
newpage
print violfreq
newpage
output printer with unsfreq.rpt
print unsfreq
newpage
output screen
remove frqchart
remove temp
newpage
clear all variables
return
*( end )


$COMMAND
BOARDING
*(*********************************************************************

    Program:             Boarding Summary Listing Program
    Author:              J D Allen and H R Lang
    Date Written:        August 1987
    Description:         This program performs a listing of the
                         BOARDING table during a specified time
                         period.  The report is sorted by date
                         before being printed.
    Tables Used:         boarding
    Temp Tables:         bosum
    Reports Used:        boarding
    Forms Used:          none

*********************************************************************)

set null " "
set error messages off
set escape on
set messages off
newpage
clear all variables
write "This program will dump all information on boardings " at 3 1
write "which are selected based on boarding date.  If you wish to +
print" at 4 1
write "out the entire database of boardings, please enter the date " +
at 5 1
write "[01/01/1900] for the starting date and todays date for the " +
at 6 1
write "ending date......................................... " +
at 7 1
set var strtdate date
set var enddate date
set var datediff to 0
set error messages on
while datediff <= 0 then
   write "Please enter starting date in the form MM/DD/YYYY:" at 9 1
   fillin strtdate using " " at 9 53
   write "Please enter ending date in the form MM/DD/YYYY:" at 11 1
   fillin enddate using " " at 11 53
   set var datediff to .enddate - .strtdate
endwhile
set error messages off
remove bosum
newpage
project bosum from boarding using all sorted by bodate where +
bodate >= .strtdate and bodate <= .enddate
compute numrows as rows from bosum
write "Now preparing report..........." at 11 1
if numrows fails or numrows = 0 then
   write "No records match date selection criteria" at 10 10
   write "Please press any key to continue" at 16 10
   pause
else
   output printer with boarding.rpt
   newpage
   print boarding
   newpage
   output screen
endif
remove bosum
newpage
clear all variables
return
*( end )


$COMMAND
BOMAJOR
*(**********************************************************************

    Program:            Boarding Major Grid Area Frequency Program
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This program performs a statistical analysis
                        of Boardings to determine percentage of cases
                        which occur in areas specified by coordinates
                        found in the MAJAREA table during a specified
                        time period.
    Tables Used:        majarea, boarding
    Temp Tables:        bosum, positfrq
    Reports Used:       bomajor
    Forms Used:         none

**********************************************************************)

set null " "
set error messages off
set escape on
set messages off
newpage
clear all variables
write "This module performs a statistical analysis of boarding " +
at 2 1
write "locations as defined by major grid areas during a specified " +
at 3 1
write "period. These major grid areas may be changed by modifying" +
at 4 1
```

```
write "the grid positions in table MAJAREA" at 5 1
set var strtdate date
set var enddate date
set var datediff to 0
set error messages on
while datediff <= 0 then
   write "Please enter starting date in the form MM/DD/YYYY:" at 8 1
   fillin strtdate using " " at 8 51
   write "Please enter ending date in the form MM/DD/YYYY:" at 10 1
   fillin enddate using " " at 10 51
   set var datediff to .enddate - .strtdate
endwhile
set error messages off
remove positfrq
remove bosum
newpage
set var northtst real
set var southtst real
set var westtst real
set var easttst real
set var gridcnt integer
project positfrq from majarea using all sorted by majgrid
expand positfrq with freqcnt integer
expand positfrq with gridpct real
newpage
write "This program will take several minutes to process the report." +
at 10 1
write "Please be patient.........." at 11 1
project bosum from boarding using bono obslat obslong where +
bodate ge .strtdate and bodate le .enddate
compute totbrds as count bono from bosum
set var gridpct real
set var temppct real
set pointer #1 grid for positfrq
while grid eq 0 then
   set var northtst to majnorth in #1
   set var southtst to majsouth in #1
   set var easttst to majeast in #1
   set var westtst to majwest in #1
   compute gridcnt as count bono from bosum where obslat < .northtst +
   and obslat >= .southtst and obslong < .westtst and +
   obslong >= .easttst
   change freqcnt to .gridcnt in #1
   set var temppct to .gridcnt / .totbrds
   if temppct fails then
      set var temppct to 0.0
   endif
   assign gridpct to .temppct x 100.0 in #1
   next #1 grid
endwhile
set var inarea integer
set var outarea integer
set var inpct real
set var outpct real
compute inarea as sum freqcnt from positfrq
set var outarea to .totbrds - .inarea
set var inpct to .inarea / .totbrds
set var outpct to .outarea / .totbrds
set var inpct to .inpct x 100.0
set var outpct to .outpct x 100.0
newpage
write "Processing report information has been completed." at 3 1
write "Now preparing report.........." at 4 1
output printer with bomajor.rpt
newpage
print bomajor
newpage
output terminal
remove positfrq
remove bosum
```

```
newpage
clear all variables
return
*( end )


$COMMAND
BOMINOR
*(************************************************************************

     Program:            Boarding Minor Grid Area Frequency Program
     Author:             J D Allen and H R Lang
     Date Written:       August 1987
     Description:        This program performs a statistical analysis
                         of Boardings to determine percentage of
                         cases which occur in a specified location
                         defined in the MINAREA table during a
                         specified time period.
     Tables Used:        boarding, minarea
     Temp Tables:        bosum, positfrq
     Reports Used:       bominor
     Forms Used:         none

************************************************************************)

set null " "
set error messages off
set escape on
set messages off
newpage
clear all variables
write "This module performs a statistical analysis of boarding " +
at 2 1
write "locations as defined by minor grid areas during a specified " +
at 3 1
write "period. These minor grid areas may be changed by modifying" +
at 4 1
write "the grid positions in table MINAREA" at 5 1
set var strtdate date
set var enddate date
set var datediff to 0
set error messages on
while datediff <= 0 then
   write "Please enter starting date in the form MM/DD/YYYY:" at 8 1
   fillin strtdate using " " at 8 51
   write "Please enter ending date in the form MM/DD/YYYY:" at 10 1
   fillin enddate using " " at 10 51
   set var datediff to .enddate - .strtdate
endwhile
set error messages off
remove positfrq
remove bosum
newpage
set var northtst real
set var southtst real
set var westtst real
set var easttst real
set var gridcnt integer
project positfrq from minarea using all sorted by mingrid
expand positfrq with freqcnt integer
expand positfrq with gridpct real
newpage
write "This program will take several minutes to process the report." +
at 10 1
write "Please be patient.........." at 11 1
project bosum from boarding using bono obslat obslong where +
bodate ge .strtdate and bodate le .enddate
compute totbrds as count bono from bosum
```

```
set var gridpct real
set var temppct real
set pointer #1 grid for positfrq
while grid eq 0 then
   set var northtst to minnorth in #1
   set var southtst to minsouth in #1
   set var easttst to mineast in #1
   set var westtst to minwest in #1
   compute gridcnt as count bono from bosum where obslat < .northtst +
   and obslat >= .southtst and obslong < .westtst and +
   obslong >= .easttst
   change freqcnt to .gridcnt in #1
   set var temppct to .gridcnt / .totbrds
   if temppct fails then
      set var temppct to 0.0
   endif
   assign gridpct to .temppct x 100.0 in #1
   next #1 grid
endwhile
set var inarea integer
set var outarea integer
set var inpct real
set var outpct real
compute inarea as sum freqcnt from positfrq
set var outarea to .totbrds - .inarea
set var inpct to .inarea / .totbrds
set var outpct to .outarea / .totbrds
set var inpct to .inpct x 100.0
set var outpct to .outpct x 100.0
newpage
write "Processing report information has been completed." at 3 1
write "Now preparing report.........." at 4 1
output printer with bominor
newpage
print bominor
newpage
output terminal
remove positfrq
remove bosum
newpage
clear all variables
return
*( end )


$COMMAND
SMBOATRP
*(************************************************************************

    Program:            Boarding Small Boat Area Frequency Program
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This program performs a statistical analysis
                        of Boardings to determine percentage of cases
                        which occur in the small boat operating as
                        defined by Coast Guard Group Monterey's
                        OPLAN during a specified time period.
    Tables Used:        boarding
    Temp Tables:        temp, bosum
    Reports Used:       smboatrp
    Forms Used:         none

************************************************************************)

set error messages off
set escape on
set messages off
newpage
```

```
clear all variables
remove temp
remove bosum
newpage
write "This module performs a statistical analysis of boarding " +
at 2 1
write "locations.  This program examines the number of boardings   " +
at 3 1
write "inside and outside of the small boat operational area during " +
at 4 1
write "a specified time period.          " at 5 1
set var strtdate date
set var enddate date
set var datediff to 0
set error messages on
while datediff <= 0 then
   write "Please enter starting date in the form MM/DD/YYYY:" at 8 1
   fillin strtdate using " " at 8 51
   write "Please enter ending date in the form MM/DD/YYYY:" at 10 1
   fillin enddate using " " at 10 51
   set var datediff to .enddate - .strtdate
endwhile
set error messages off
newpage
write "This program will take several minutes to process the report." +
at 10 1
write "Please be patient.........." at 11 1
project bosum from boarding using bono obslat obslong where +
bodate >= .strtdate and bodate <= .enddate
compute totbrds as count bono from bosum
set var inside to 0
set var outside to 0
set pointer #1 n1 for bosum
while n1 eq 0 then
   while n1 eq 0 then
   set var obslat to obslat in #1
   set var obslong to obslong in #1
      if obslat < 36.100 then
         set var outside to .outside + 1
         break
      endif
      if obslong < 121.40 then
         set var outside to .outside + 1
         break
      endif
      if obslat >= 36.100 and obslat <= 36.180 and obslong >= 121.400 +
      and obslong <= 122.200 then
         set var inside to .inside + 1
         break
      endif
      if obslat >= 36.180 and obslat <= 36.370 and obslong >= 121.400 +
      and obslong <= 122.250 then
         set var inside to .inside + 1
         break
      endif
      if obslat >= 36.370 and obslat <= 36.480 and obslong >= 121.400 +
      and obslong <= 122.300 then
         set var inside to .inside + 1
         break
      endif
      if obslat >= 36.480 and obslat <= 36.530 and obslong >= 121.400 +
      and obslong <= 122.350 then
         set var inside to .inside + 1
         break
      endif
      if obslat >= 36.530 and obslat <= 36.570 and obslong >= 121.400 +
      and obslong <= 122.400 then
         set var inside to .inside + 1
         break
      endif
      if obslat >= 36.570 and obslat <= 37.000 and obslong >= 121.400 +
      and obslong <= 122.450 then
         set var inside to .inside + 1
         break
     endif
     if obslat >= 37.000 and obslat <= 37.050 and obslong >= 121.400 +
     and obslong <= 122.450 then
         set var inside to .inside + 1
         break
     endif
     if obslat >= 37.050 and obslat <= 37.100 and obslong >= 121.400 +
     and obslong <= 122.300 then
         set var inside to .inside + 1
         break
     endif
     set var outside to .outside + 1
     break
  endwhile
  next #1 n1
endwhile
newpage
define
attributes
tmp text 2
relations
temp with tmp
end
load temp
   "XX"
end
write "Processing report information has been completed." at 3 1
write "Now preparing report..........." at 4 1
output printer with smboatrp.rpt
newpage
print smboatrp
newpage
output terminal
remove temp
remove bosum
newpage
clear all variables
return
*( end )


$COMMAND
CMERPT
*(*********************************************************************

    Program:          Courtesy Motorboat Examinations (CME's)
                      Statistical Analysis
    Author:           J D Allen and H R Lang
    Date Written:     August 1987
    Description:      This program performs a statistical analysis
                      of vessels that were boarded to determine
                      the frequency of CME's and their year of
                      issue during a specified time period.
    Tables Used:      boarding, vessel
    Temp Tables:      bosum, vslsum, bovslsum, temp
    Reports Used:     cmerpt
    Forms Used:       none

**********************************************************************)

set error messages off
set escape on
set messages off
newpage
```

```
clear all variables
write "This program will perform statistical reports on" at 3 1
write "vessels that were boarded during a specified time with" at 4 1
write "USCG Auxiliary Courtesy Motorboat Examinations CMEs Decals" +
at 5 1
set var strtdate date
set var enddate date
set var datediff to 0
set error messages on
while datediff <= 0 then
   write "Please enter starting date in the form MM/DD/YYYY:" at 8 1
   fillin strtdate using " " at 8 53
   write "Please enter ending date in the form MM/DD/YYYY:" at 10 1
   fillin enddate using " " at 10 53
   set var datediff to .enddate - .strtdate
endwhile
set error messages off
remove bosum
remove vslsum
remove bovslsum
newpage
project bosum from boarding using bono bovhin where +
bodate ge .strtdate and bodate le .enddate
project vslsum from vessel using cmedecal cmeyear vslhin
join bosum using bovhin with vslsum using vslhin forming bovslsum +
where eq
compute totbrds as count bono from bovslsum
compute cmecount as count cmedecal from bovslsum where cmedecal = Y
if cmecount >= 1 and totbrds >= 1 then
   compute before70 as count cmedecal from bovslsum where cmedecal = Y +
   and cmeyear <= 1970
   compute yr71to75 as count cmedecal from bovslsum where cmedecal = Y +
   and cmeyear >= 1971 and cmeyear <= 1975
   compute yr76to79 as count cmedecal from bovslsum where cmedecal = Y +
   and cmeyear >= 1976 and cmeyear <= 1979
   compute yr80to81 as count cmedecal from bovslsum where cmedecal = Y +
   and cmeyear >= 1980 and cmeyear <= 191
   compute yr82to83 as count cmedecal from bovslsum where cmedecal = Y +
   and cmeyear >= 1982 and cmeyear <= 1983
   compute yr84to85 as count cmedecal from bovslsum where cmedecal = Y +
   and cmeyear >= 1984 and cmeyear <= 1985
   compute yr86to87 as count cmedecal from bovslsum where cmedecal = Y +
   and cmeyear >= 1986 and cmeyear <= 1987
   compute yr88to89 as count cmedecal from bovslsum where cmedecal = Y +
   and cmeyear >= 1988 and cmeyear <= 1989
   compute after90 as count cmedecal from bovslsum where cmedecal = Y +
   and cmeyear >= 1990
   set var pctlt70 real
   set var pct7175 real
   set var pct7679 real
   set var pct8081 real
   set var pct8283 real
   set var pct8485 real
   set var pct8687 real
   set var pct8889 real
   set var pctgt90 real
   if totbrds fails or totbrds = 0 then
      set var totbrds to 1
   endif
   set var pctlt70 to .before70 / .totbrds
   set var pct7175 to .yr71to75 / .totbrds
   set var pct7679 to .yr76to79 / .totbrds
   set var pct8081 to .yr80to81 / .totbrds
   set var pct8283 to .yr82to83 / .totbrds
   set var pct8485 to .yr84to85 / .totbrds
   set var pct8687 to .yr86to87 / .totbrds
   set var pct8889 to .yr88to89 / .totbrds
   set var pctgt90 to .after90 / .totbrds
   set var vpct70 real
   set var vpct71 real
   set var vpct76 real
   set var vpct80 real
   set var vpct82 real
   set var vpct84 real
   set var vpct86 real
   set var vpct88 real
   set var vpct90 real
   if cmecount = 0 or cmecount fails then
      set var cmecount to 1
   endif
   set var vpct70 to .before70 / .cmecount
   set var vpct71 to .yr71to75 / .cmecount
   set var vpct76 to .yr76to79 / .cmecount
   set var vpct80 to .yr80to81 / .cmecount
   set var vpct82 to .yr82to83 / .cmecount
   set var vpct84 to .yr84to85 / .cmecount
   set var vpct86 to .yr86to87 / .cmecount
   set var vpct88 to .yr88to89 / .cmecount
   set var vpct90 to .after90 / .cmecount
   set var pctcme real
   set var pctcme to .cmecount / .totbrds
   remove temp
   newpage
   write "Processing of statistical information completed." at 10 1
   write "Now preparing report............" at 11 1
     *(creating temporary relation for printing purposes)
   define
   attributes
   tmp text 2
   relations
   temp with tmp
   end
   load temp
      "XX"
   end
   output printer with cmerpt.rpt
   newpage
   print cmerpt
   newpage
   output screen
else
   newpage
   write "No records match your selection criteria" at 10 15
   write "Please press any key to continue" at 15 20
   pause
   newpage
endif
remove bosum
remove vslsum
remove bovslsum
remove temp
clear all variables
return
*( end )


$COMMAND
DAYNIGHT
*(**********************************************************************

    Program:              Daytime vs. Nighttime Frequency Analysis
    Author:               J D Allen and H R Lang
    Date Written:         August 1987
    Description:          This program performs a statistical analysis
                          of SAR cases and Boardings to determine
                          percentage of cases which occur in the
                          daytime vs those that occur at night during
                          a specified time period.
    Tables Used:          sar, boarding
    Temp Tables:       sarsum, bosum, temp
    Reports Used:      daynight
    Forms Used:        none

*************************************************************************)

set null " "
set error messages off
set escape on
set messages off
clear all variables
remove sarsum
newpage
write "This program performs a statistical analysis of Search &" at 2 1
write "Rescue cases and Boardings to determine the percentage of " +
at 3 1
write "cases which occurred during the day and at night during a " +
at 4 1
write "specified time period. " at 5 1
set var strtdate date
set var enddate date
set var datediff to 0
set error messages on
while datediff <= 0 then
   write "Please enter starting date in the form MM/DD/YYYY:" at 8 1
   fillin strtdate using " " at 8 51
   write "Please enter ending date in the form MM/DD/YYYY:" at 10 1
   fillin enddate using " " at 10 51
   set var datediff to .enddate - .strtdate
endwhile
set error messages off
project sarsum from sar using ssunrise ssunset sartime sardate where +
sardate >= .strtdate and sardate <= .enddate and ssunrise exists and +
sartime exists
compute daysar as count sartime from sarsum where sartime gea +
ssunrise and sartime lta ssunset
compute nitesar as count sartime from sarsum where sartime lta +
ssunrise or sartime gea ssunset
set var totsars to .daysar + .nitesar
set var pctdays real
set var pctnites real
set var pctdays to .daysar / .totsars
set var pctnites to .nitesar / .totsars
set var pctdays to .pctdays x 100.0
set var pctnites to .pctnites x 100.0
remove sarsum
remove bosum
newpage
project bosum from boarding using bsunset bsunrise botime bodate where +
bodate >= .strtdate and bodate <= .enddate and bsunrise exists and +
botime exists
compute daybrd as count botime from bosum where botime gea bsunrise +
and botime lta bsunset
compute nitebrd as count botime from bosum where botime lta bsunrise +
or botime gea bsunset
set var totbrds to .daybrd + .nitebrd
set var pctdayb real
set var pctniteb real
set var pctdayb to .daybrd / .totbrds
set var pctniteb to .nitebrd / .totbrds
set var pctdayb to .pctdayb x 100.0
set var pctniteb to .pctniteb x 100.0
remove bosum
remove temp
newpage
define
attributes
tmp text 2
relations
```

```
temp with tmp
end
load temp
   "XX"
end
newpage
write "Preparing report....."
output printer with daynight.rpt
newpage
print daynight
newpage
output screen
remove sarsum
remove bosum
remove temp
newpage
return
*( end )
```

APPENDIX L
SARRPTS SOURCE CODE

```
$COMMAND
SAR_MAIN
*(*********************************************************************

                     US COAST GUARD GROUP MONTEREY
         SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                     INTELLIGENCE SYSTEM (SARELTIS)

               SAR REPORTS MENU and APPLICATION PROGRAMS

                 Author:    Jon D. Allen, LT, USCG
                            Heidi R. Lang, LT, USN
                            AUGUST 1987

**********************************************************************)

set error messages off
set messages off
set escape off
clear all variables
newpage
set var pick7 integer
choose pick7 from sar_menu in sarrpts.apx
if pick7 eq 0 then
   quit to restsar.cmd
endif
if pick7 eq 1 then
   run IMMEDSAR in sarrpts.apx
   quit to restsar.cmd
endif
if pick7 eq 2 then
   run SARMINOR in sarrpts.apx
   quit to restsar.cmd
endif
if pick7 eq 3 then
   run SARMAJOR in sarrpts.apx
   quit to restsar.cmd
endif
if pick7 eq 4 then
   run SARSMBT in sarrpts.apx
   quit to restsar.cmd
endif
if pick7 eq 5 then
   run SARDUMP in sarrpts.apx
   quit to restsar.cmd
endif
if pick7 eq 6 then
   run SARTIMES in sarrpts.apx
   quit to restsar.cmd
endif
if pick7 eq 7 then
   run SARDAY in sarrpts.apx
   quit to restsar.cmd
endif
if pick7 eq 8 then
   run DAYNIGHT in sarrpts.apx
   quit to restsar.cmd
endif
if pick7 eq 9 then
```

```
   quit to restrpts.cmd
endif
quit to restsar.cmd
return
$menu
sar_menu
column SARELTIS --- SAR REPORTS
ANALYSIS OF IMMEDIATE DISTRESS SAR CASES
MINOR AREA (MINAREA table) LOCATION ANALYSIS
MAJOR AREA (MAJAREA table) LOCATION ANALYSIS
INSIDE/OUTSIDE SMALL BOAT OPAREA LOCATION ANALYSIS
SAR TABLE SUMMARY LISTING
SAR UNIT TIME EXPENDED DURING SAR CASES
DAY OF THE WEEK,  MONTH,  AND YEAR ANALYSIS
SAR CASE ANALYSIS:  DAYTIME/NIGHTTIME FREQUENCY
RETURN TO REPORTS MENU
*( end )


$COMMAND
SARMAJOR
*(*********************************************************************

    Program:            SAR Major Grid Area Frequency Analysis
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This program performs a statistical analysis
                        of SAR cases to determine percentage of
                        cases which occur in specified locations in
                        the table MAJAREA during a specified time
                        period.
    Tables Used:        majarea, sar
    Temp Tables:        sarsum, positfrq
    Reports Used:       sarmajor
    Forms Used:         none

**********************************************************************)

set null " "
set error messages off
set messages off
set escape on
newpage
clear all variables
write "This module performs a statistical analysis of SAR case " at 2 1
write "locations as defined by major grid areas during a specified " +
at 3 1
write "period. These major grid areas may be changed by modifying" +
at 4 1
write "the grid positions in table MAJAREA" at 5 1
set var strtdate date
set var enddate date
set var datediff to 0
set error messages on
while datediff <= 0 then
   write "Please enter starting date in the form MM/DD/YYYY:" at 8 1
   fillin strtdate using " " at 8 51
   write "Please enter ending date in the form MM/DD/YYYY:" at 10 1
   fillin enddate using " " at 10 51
   set var datediff to .enddate - .strtdate
endwhile
set error messages off
remove positfrq
remove sarsum
newpage
set var northtst real
set var southtst real
set var westtst real
set var easttst real
set var gridcnt integer
project positfrq from majarea using all sorted by majgrid
expand positfrq with freqcnt integer
expand positfrq with gridpct real
newpage
write "This program will take several minutes to process the report." +
at 10 1
write "Please be patient.........." at 11 1
project sarsum from sar using folderno sarlat sarlong where +
sardate ge .strtdate and sardate le .enddate
compute totsars as count folderno from sarsum
compute numrows as rows from sarsum
set var gridpct real
set var temppct real
set pointer #1 grid for positfrq
while grid eq 0 then
   set var northtst to majnorth in #1
   set var southtst to majsouth in #1
   set var easttst to majeast in #1
   set var westtst to majwest in #1
   compute gridcnt as count folderno from sarsum where +
   sarlat < .northtst and sarlat >= .southtst and sarlong < .westtst +
   and sarlong >= .easttst
   change freqcnt to .gridcnt in #1
   set var temppct to .gridcnt / .totsars
   if temppct fails then
      set var temppct to 0.0
   endif
   assign gridpct to .temppct x 100.0 in #1
   next #1 grid
endwhile
set var inarea integer
set var outarea integer
set var inpct real
set var outpct real
compute inarea as sum freqcnt from positfrq
set var outarea to .totsars - .inarea
set var inpct to .inarea / .totsars
set var outpct to .outarea / .totsars
set var inpct to .inpct x 100.0
set var outpct to .outpct x 100.0
newpage
write "Processing report information has been completed." at 3 1
write "Now preparing report..........." at 4 1
if numrows fails or numrows = 0 then
   newpage
   write "No records exist in the database matching your selection +
criteria" at 10 1
   write "Please press any key to return to menu" at 16 10
   pause
else
   output printer with sarmajor.rpt
   newpage
   print sarmajor
   newpage
   output terminal
endif
remove positfrq
remove sarsum
newpage
clear all variables
return
*( end )


$COMMAND
SARMINOR
*(**********************************************************************
    Program:           SAR Minor Grid Area Frequency Analysis
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program performs a statistical analysis
                       of SAR cases to determine percentage of
                       cases which occur in locations specified by
                       coordinates contained in the table MINAREA
                       during a specified time period.
    Tables Used:       sar, minarea
    Temp Tables:       sarsum, positfrq
    Reports Used:      sarminor
    Forms Used:        none

************************************************************************)

set null " "
set error messages off
set messages off
set escape on
newpage
clear all variables
write "This module performs a statistical analysis of SAR case " at 2 1
write "locations as defined by minor grid areas during a specified " +
at 3 1
write "period. These major grid areas may be changed by modifying" +
at 4 1
write "the grid positions in table MINAREA" at 5 1
set var strtdate date
set var enddate date
set var datediff to 0
set error messages on
while datediff <= 0 then
   write "Please enter starting date in the form MM/DD/YYYY:" at 8 1
   fillin strtdate using " " at 8 51
   write "Please enter ending date in the form MM/DD/YYYY:" at 10 1
   fillin enddate using " " at 10 51
   set var datediff to .enddate - .strtdate
endwhile
set error messages off
remove positfrq
remove sarsum
newpage
set var northtst real
set var southtst real
set var westtst real
set var easttst real
set var gridcnt integer
project positfrq from minarea using all sorted by mingrid
expand positfrq with freqcnt integer
expand positfrq with gridpct real
newpage
write "This program will take several minutes to process the report." +
at 10 1
write "Please be patient.........." at 11 1
project sarsum from sar using folderno sarlat sarlong where +
sardate ge .strtdate and sardate le .enddate
compute totsars as count folderno from sarsum
compute numrows as rows from sarsum
set var gridpct real
set var temppct real
set pointer #1 grid for positfrq
while grid eq 0 then
   set var northtst to minnorth in #1
   set var southtst to minsouth in #1
   set var easttst to mineast in #1
   set var westtst to minwest in #1
   compute gridcnt as count folderno from sarsum where +
   sarlat < .northtst and sarlat >= .southtst and sarlong < .westtst +
   and sarlong >= .easttst
   change freqcnt to .gridcnt in #1
   set var temppct to .gridcnt / .totsars
   if temppct fails then
      set var temppct to 0.0
   endif
   assign gridpct to .temppct x 100.0 in #1
   next #1 grid
endwhile
set var inarea integer
set var outarea integer
set var inpct real
set var outpct real
compute inarea as sum freqcnt from positfrq
set var outarea to .totsars - .inarea
set var inpct to .inarea / .totsars
set var outpct to .outarea / .totsars
set var inpct to .inpct x 100.0
set var outpct to .outpct x 100.0
newpage
write "Processing report information has been completed." at 3 1
write "Now preparing report.........." at 4 1
if numrows fails or numrows = 0 then
   newpage
   write "No records exist in the database matching your selection +
criteria" at 10 1
   write "Please press any key to return to menu" at 16 10
   pause
else
   output printer with sarminor.rpt
   newpage
   print sarminor
   newpage
   output terminal
endif
remove positfrq
remove sarsum
newpage
clear all variables
return
*( end )


$COMMAND
SARSMBT
*(*********************************************************************

    Program:            SAR Case Small Boat Area Frequency Program
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This program performs a statistical analysis
                        of SAR Cases to determine percentage of
                        cases which occur in the small boat operating
                        as defined by Coast Guard Group Monterey's
                        OPLAN during a specified time period.
    Tables Used:        sar
    Temp Tables:        temp, sarsum
    Reports Used:       sarsmbt
    Forms Used:         none

*********************************************************************)

set null " "
set error messages off
set messages off
set escape on
newpage
clear all variables
```

```
remove temp
remove sarsum
newpage
write "This module performs a statistical analysis of SAR Case " at 2 1
write "locations.  This program examines the number of SAR Cases   " +
at 3 1
write "inside and outside of the small boat operational area during " +
at 4 1
write "a specified time period.          " at 5 1
set var strtdate date
set var enddate date
set var datediff to 0
set error messages on
while datediff <= 0 then
   write "Please enter starting date in the form MM/DD/YYYY:" at 8 1
   fillin strtdate using " " at 8 51
   write "Please enter ending date in the form MM/DD/YYYY:" at 10 1
   fillin enddate using " " at 10 51
   set var datediff to .enddate - .strtdate
endwhile
set error messages off
newpage
write "This program will take several minutes to process the report." +
at 10 1
write "Please be patient.........." at 11 1
project sarsum from sar using folderno sarlat sarlong where +
sardate >= .strtdate and sardate <= .enddate
compute totsars as count folderno from sarsum
compute numrows as rows from sarsum
set var inside to 0
set var outside to 0
set pointer #1 n1 for sarsum
while n1 eq 0 then
   while n1 eq 0 then
   set var sarlat to sarlat in #1
   set var sarlong to sarlong in #1
      if sarlat < 36.100 then
         set var outside to .outside + 1
         break
      endif
      if sarlong < 121.40 then
         set var outside to .outside + 1
         break
      endif
      if sarlat >= 36.100 and sarlat <= 36.180 and sarlong >= 121.400 +
      and sarlong <= 122.200 then
         set var inside to .inside + 1
         break
      endif
      if sarlat >= 36.180 and sarlat <= 36.370 and sarlong >= 121.400 +
      and sarlong <= 122.250 then
         set var inside to .inside + 1
         break
      endif
      if sarlat >= 36.370 and sarlat <= 36.480 and sarlong >= 121.400 +
      and sarlong <= 122.300 then
         set var inside to .inside + 1
         break
      endif
      if sarlat >= 36.480 and sarlat <= 36.530 and sarlong >= 121.400 +
      and sarlong <= 122.350 then
         set var inside to .inside + 1
         break
      endif
      if sarlat >= 36.530 and sarlat <= 36.570 and sarlong >= 121.400 +
      and sarlong <= 122.400 then
         set var inside to .inside + 1
         break
      endif
      if sarlat >= 36.570 and sarlat <= 37.000 and sarlong >= 121.400 +
      and sarlong <= 122.450 then
         set var inside to .inside + 1
         break
     endif
     if sarlat >= 37.000 and sarlat <= 37.050 and sarlong >= 121.400 +
     and sarlong <= 122.450 then
         set var inside to .inside + 1
         break
     endif
     if sarlat >= 37.050 and sarlat <= 37.100 and sarlong >= 121.400 +
     and sarlong <= 122.300 then
         set var inside to .inside + 1
         break
     endif
     set var outside to .outside + 1
     break
  endwhile
  next #1 nl
endwhile
set var inpcttmp real
set var outpcttm real
set var inpct real
set var outpct real
set var totsars real
set var inside real
set var outside real
set var inpcttmp to .inside / .totsars
set var outpcttm to .outside / .totsars
set var inpct to .inpcttmp x 100.0
set var outpct to .outpcttm x 100.0
set var totsars integer
set var inside integer
set var outside integer
newpage
define
attributes
tmp text 2
relations
temp with tmp
end
load temp
   "XX"
end
write "Processing report information has been completed." at 3 1
write "Now preparing report..........." at 4 1
if numrows fails or numrows = 0 then
   newpage
   write "No records exist in the database matching your selection +
   criteria" at 10 1
   write "Please press any key to return to menu" at 16 10
   pause
else
   output printer with sarsmbt.rpt
   newpage
   print sarsmbt
   newpage
   output terminal
endif
remove temp
remove sarsum
newpage
clear all variables
return
*( end )


$COMMAND
SARTIMES
*(***********************************************************************
    Program:           SAR Case Units Time Expended Frequency
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program performs a statistical analysis
                       of SAR cases to determine total time
                       expended, time to being on scene, time to
                       being alongside, time involved in searches,
                       and time involved in towing.  These times are
                       analyzed to produce totals, averages,
                       minimums, and maximums for a specific time
                       period.
    Tables Used:       sarunits
    Temp Tables:       sartime1, sartime2, sartime3, sartime4,
                       sartime5, temp
    Reports Used:      sartimes
    Forms Used:        none

***********************************************************************)

set null " "
set error messages off
set messages off
set escape on
newpage
clear all variables
write "This program will perform statistical reports on" at 3 1
write "SAR cases that occurred during a specified time period." at 4 1
write "Total sar unit time expended,  time to on scene,  time " at 5 1
write "to alongside distressed vessel,  time expended in searches," +
at 6 1
write "and time engaged in towing will be analyzed." at 7 1
write "Since this program can take 5 to 10 minutes to run,  you may" +
at 12 1
write "press the escape [ESC] key to exit to the RBase 5000 command" +
at 13 1
write "prompt where can either restart the menu system by typing:" +
at 14 1
write "                    RUN STARTUP.BIN" at 16 1
write "or exitting the system by typing:" at 18 1
write "                    EXIT" at 20 1
write "Press any key to continue " at 23 10
pause
newpage
set var strtdate date
set var enddate date
set var datediff to 0
set error messages on
while datediff <= 0 then
   write "Please enter starting date in the form MM/DD/YYYY:" at 10 1
   fillin strtdate using " " at 10 53
   write "Please enter ending date in the form MM/DD/YYYY:" at 12 1
   fillin enddate using " " at 12 53
   set var datediff to .enddate - .strtdate
endwhile
set error messages off
remove sartime1
newpage
project sartime1 from sarunits using uwtime uwdate ostime osdate +
foldno where uwtime exists and uwdate exists and ostime exists and +
osdate exists and uwdate lea osdate and uwdate >= .strtdate and +
sutdate <= .enddate
compute ostimes as count foldno from sartime1
set var ossum real
set var timehour real
set var datehour real
set var test real
set var timeadd real
set var minostim real
set var maxostim real
set var ossum to 0.0
set var minostim to 1000.0
set var maxostim to 0.0
write "Now computing sar time statistics" at 10 15
set pointer #1 n1 for sartime1
while n1 eq 0 then
   set var uwtime to uwtime in #1
   set var uwdate to uwdate in #1
   set var ostime to ostime in #1
   set var osdate to osdate in #1
   if uwdate = .osdate then
      set var timediff to .ostime - .uwtime
      set var timehour to .timediff / 3600.0
      *(converts seconds to hours)
      set var test to .timehour
      set var ossum to .ossum + .timehour
   else
      set var datediff to .osdate - .uwdate
      set var datehour to .datediff x 24.0
      *(converts days to hours)
      set var timediff to .ostime - .uwtime
      set var timehour to .timediff / 3600.0
      *(converts seconds to hours)
      set var timeadd to .timehour + .datehour
      set var test to .timeadd
      set var ossum to .ossum + .timeadd
   endif
   if test <= .minostim then
      set var minostim to .test
   endif
   if test >= .maxostim then
      set var maxostim to .test
   endif
   next #1 n1
endwhile
remove sartime1
remove sartime2
newpage
project sartime2 from sarunits using uwtime uwdate sutmoor sutdate +
foldno where uwtime exists and uwdate exists and sutmoor exists and +
sutdate exists and uwdate lea sutdate and uwdate >= .strtdate and +
sutdate <= .enddate
compute tottimes as count foldno from sartime2
set var totsum real
set var mintttim real
set var maxtttim real
set var totsum to 0.0
set var mintttim to 1000.0
set var maxtttim to 0.0
set pointer #1 n1 for sartime2
while n1 eq 0 then
   set var uwtime to uwtime in #1
   set var uwdate to uwdate in #1
   set var sutmoor to sutmoor in #1
   set var sutdate to sutdate in #1
   if uwdate = .sutdate then
      set var timediff to .sutmoor - .uwtime
      set var timehour to .timediff / 3600.0
      *(converts seconds to hours)
      set var test to .timehour
      set var totsum to .totsum + .timehour
   else
      set var datediff to .sutdate - .uwdate
      set var datehour to .datediff x 24.0
      *(converts days to hours)
      set var timediff to .sutmoor - .uwtime
      set var timehour to .timediff / 3600.0
      *(converts seconds to hours)
      set var timeadd to .timehour + .datehour
      set var test to .timeadd
      set var totsum to .totsum + .timeadd
   endif
   if test <= .mintttim then
      set var mintttim to .test
   endif
   if test >= .maxtttim then
      set var maxtttim to .test
   endif
   next #1 n1
endwhile
remove sartime2
remove sartime3
newpage
project sartime3 from sarunits using uwtime uwdate alngtime alngdate +
foldno where uwtime exists and uwdate exists and alngtime exists and +
alngdate exists and uwdate lea alngdate and uwdate >= .strtdate and +
sutdate <= .enddate
compute algtimes as count foldno from sartime3
set var algsum real
set var minaltim real
set var maxaltim real
set var algsum to 0.0
set var minaltim to 1000.0
set var maxaltim to 0.0
set pointer #1 n1 for sartime3
while n1 eq 0 then
   set var uwtime to uwtime in #1
   set var uwdate to uwdate in #1
   set var algtime to alngtime in #1
   set var algdate to alngdate in #1
   if uwdate = .algdate then
      set var timediff to .algtime - .uwtime
      set var timehour to .timediff / 3600.0
      *(converts seconds to hours)
      set var test to .timehour
      set var algsum to .algsum + .timehour
   else
      set var datediff to .algdate - .uwdate
      set var datehour to .datediff x 24.0
      *(converts days to hours)
      set var timediff to .algtime - .uwtime
      set var timehour to .timediff / 3600.0
      *(converts seconds to hours)
      set var timeadd to .timehour + .datehour
      set var test to .timeadd
      set var algsum to .algsum + .timeadd
   endif
   if test <= .minaltim then
      set var minaltim to .test
   endif
   if test >= .maxaltim then
      set var maxaltim to .test
   endif
   next #1 n1
endwhile
remove sartime3
remove sartime4
newpage
project sartime4 from sarunits using ostime osdate alngtime alngdate +
foldno where ostime exists and osdate exists and alngtime exists and +
alngdate exists and osdate lea alngdate and uwdate >= .strtdate and +
sutdate <= .enddate
compute schtimes as count foldno from sartime4
set var schsum real
set var minshtim real
set var maxshtim real
set var schsum to 0.0
set var minshtim to 1000.0
set var maxshtim to 0.0
set pointer #1 n1 for sartime4
while n1 eq 0 then
   set var ostime to ostime in #1
   set var osdate to osdate in #1
   set var algtime to alngtime in #1
   set var algdate to alngdate in #1
   if osdate = .algdate then
      set var timediff to .algtime - .ostime
      set var timehour to .timediff / 3600.0
      *(converts seconds to hours)
      set var test to .timehour
      set var schsum to .schsum + .timehour
   else
      set var datediff to .algdate - .osdate
      set var datehour to .datediff x 24.0
      *(converts days to hours)
      set var timediff to .algtime - .ostime
      set var timehour to .timediff / 3600.0
      *(converts seconds to hours)
      set var timeadd to .timehour + .datehour
      set var test to .timeadd
      set var schsum to .schsum + .timeadd
   endif
   if test <= .minshtim then
      set var minshtim to .test
   endif
   if test >= .maxshtim then
      set var maxshtim to .test
   endif
   next #1 n1
endwhile
remove sartime4
remove sartime5
newpage
project sartime5 from sarunits using towtime towdate dvmoored dvmdate +
foldno where towtime exists and towdate exists and dvmoored exists +
and dvmdate exists and towdate lea dvmdate and uwdate >= .strtdate +
and sutdate <= .enddate
compute towtimes as count foldno from sartime5
set var towsum real
set var mintwtim real
set var maxtwtim real
set var towsum to 0.0
set var mintwtim to 1000.0
set var maxtwtim to 0.0
set pointer #1 n1 for sartime5
while n1 eq 0 then
   set var towtime to towtime in #1
   set var towdate to towdate in #1
   set var dvmtime to dvmoored in #1
   set var dvmdate to dvmdate in #1
   if towdate = .dvmdate then
      set var timediff to .dvmtime - .towtime
      set var timehour to .timediff / 3600.0
      *(converts seconds to hours)
      set var test to .timehour
      set var towsum to .towsum + .timehour
   else
      set var datediff to .dvmdate - .towdate
      set var datehour to .datediff x 24.0
      *(converts days to hours)
      set var timediff to .dvmtime - .towtime
      set var timehour to .timediff / 3600.0
      *(converts seconds to hours)
      set var timeadd to .timehour + .datehour
      set var test to .timeadd
      set var towsum to .towsum + .timeadd
   endif
   if test <= .mintwtim then
      set var mintwtim to .test
   endif
   if test >= .maxtwtim then
      set var maxtwtim to .test
   endif
   next #1 n1
endwhile
remove sartime5
newpage
set var totavg real
set var oscavg real
set var algavg real
set var schavg real
set var towavg real
set var totavg to .totsum / .tottimes
set var oscavg to .ossum / .ostimes
set var algavg to .algsum / .algtimes
set var schavg to .schsum / .schtimes
set var towavg to .towsum / .towtimes
remove temp
define
attributes
tmp text 2
relations
temp with tmp
end
load temp
   "XX"
end
newpage
write "Now preparing report"  at 10 15
output printer with sartimes.rpt
newpage
print sartimes
newpage
output screen
remove temp
clear all variables
set escape off
return
*( end )


$COMMAND
SARDAY
*(**********************************************************************

    Program:          SAR Days, SAR Months, SAR Years Frequency
                      Analysis
    Author:           J D Allen and H R Lang
    Date Written:     August 1987
    Description:      This program performs a statistical analysis
                      of SAR cases to determine percentage of
                      cases which occurred on specific days of the
                      week, months of the year and years during a
                      specified time period.
    Tables Used:      sar
    Temp Tables:      sarsum1, sarstat, sarsum, temp
    Reports Used:     sardates, sardays
    Forms Used:       none

**********************************************************************)

set null " "
set error messages off
set messages off
set escape on
newpage
clear all variables
```

```
remove sarsum1
write "This program will perform a statistical report on" at 3 1
write "days of the week,  months of the year,  and individual" at 4 1
write "years that SAR cases have occurred.    " at 5 1
write "This program will evaluate statistics starting from 1980." at 8 1
write "Please indicate how many years of data you wish to analyze." +
at 9 1
set var numyears integer
set var numyears to 999999
set error messages on
while numyears <= 0 or numyears > 32 then
                              *(system will only do stats until 2012)
   fillin numyears using "Number of years: " at 12 20
   if numyears <= 0 or numyears > 32 then
      write "You must enter a number between 1 and 32" at 15 20
      write "Please reenter." at 16 30
   endif
endwhile
set var strtdate date
set var enddate date
set var strtdate to 01/01/1980
set var count integer
set var count to 1
set var daysadd integer
set var daysadd to 0
while count <= .numyears then
   set var yrtest1 to .numyears / 4
   set var yrtest2 to .yrtest1 x 4
   set var yrtest to .numyears - .yrtest2
   if yrtest = 0 then
      set var daysadd to .daysadd + 366
   else
      set var daysadd to .daysadd + 365
   endif
   set var count to .count + 1
endwhile
set var enddate to .strtdate + .daysadd
newpage
set error messages off
write "Since this program can take 5 to 10 minutes to run,  you may" +
at 12 1
write "press the escape [ESC] key to exit to the RBase 5000 command" +
at 13 1
write "prompt where can either restart the menu system by typing:" +
at 14 1
write "                    RUN STARTUP.BIN" at 16 1
write "or exitting the system by typing:" at 18 1
write "                    EXIT" at 20 1
write "Press any key to continue" at 23 10
pause
newpage
project sarsum1 from sar using sardate sarday folderno sorted by +
sardate where sardate >= .strtdate and sardate <= .enddate
compute numrows as rows from sarsum1
set var begndate date
set var stopdate date
set var begndate to 01/01/1980
set var stopdate to 12/31/1980
set var count integer
set var count to 1
set var saryear integer
set var saryear to 1979
remove sarstat
set var sarnum integer
define
attributes
saryear integer
sarnum integer
relations
sarstat with saryear sarnum
```

```
end
while count <= .numyears then
   compute numsars as count sardate from sarsum1 where +
   sardate >= .begndate and sardate <= .stopdate
   set var saryear to .saryear + 1
   if numsars fails then
      set var sarnum to 0
   else
      set var sarnum to .numsars
   endif
   load sarstat
      .saryear .sarnum
   end
   set var yrtest1 to .saryear / 4
   set var yrtest2 to .yrtest1 x 4
   set var yrtest to .saryear - .yrtest2
   if yrtest = 0 then
      set var begndate to .begndate + 366
      set var stopdate to .stopdate + 366
   else
      set var begndate to .begndate + 365
      set var stopdate to .stopdate + 365
   endif
   set var count to .count + 1
endwhile
set var jansar integer
set var count integer
set var jandate1 date
set var jandate2 date
set var febsar integer
set var febdate1 date
set var febdate2 date
set var marsar integer
set var mardate1 date
set var mardate2 date
set var aprsar integer
set var aprdate1 date
set var aprdate2 date
set var maysar integer
set var maydate1 date
set var maydate2 date
set var junsar integer
set var jundate1 date
set var jundate2 date
set var julsar integer
set var juldate1 date
set var juldate2 date
set var augsar integer
set var augdate1 date
set var augdate2 date
set var sepsar integer
set var sepdate1 date
set var sepdate2 date
set var octsar integer
set var octdate1 date
set var octdate2 date
set var novsar integer
set var novdate1 date
set var novdate2 date
set var decsar integer
set var decdate1 date
set var decdate2 date
set var jansar to 0
set var febsar to 0
set var marsar to 0
set var aprsar to 0
set var maysar to 0
set var junsar to 0
set var julsar to 0
set var augsar to 0
set var sepsar to 0
set var octsar to 0
set var novsar to 0
set var decsar to 0
set var jandate1 to 01/01/1980
set var jandate2 to 01/31/1980
set var febdate1 to 02/01/1980
set var febdate2 to 02/28/1980
set var mardate1 to 03/01/1980
set var mardate2 to 03/31/1980
set var aprdate1 to 04/01/1980
set var aprdate2 to 04/30/1980
set var maydate1 to 05/01/1980
set var maydate2 to 05/31/1980
set var jundate1 to 06/01/1980
set var jundate2 to 06/30/1980
set var juldate1 to 07/01/1980
set var juldate2 to 07/31/1980
set var augdate1 to 08/01/1980
set var augdate2 to 08/31/1980
set var sepdate1 to 09/01/1980
set var sepdate2 to 09/30/1980
set var octdate1 to 10/01/1980
set var octdate2 to 10/31/1980
set var novdate1 to 11/01/1980
set var novdate2 to 11/30/1980
set var decdate1 to 12/01/1980
set var decdate2 to 12/31/1980
set var count to 1
newpage
write "Now performing statistical analysis" at 10 10
while count <= .numyears then
   compute janrows as count sardate from sarsum1 where +
   sardate >= .jandate1 and sardate <= .jandate2
   if janrows exists then
      set var jansar to .jansar + .janrows
   endif
   compute febrows as count sardate from sarsum1 where +
   sardate >= .febdate1 and sardate <= .febdate2
   if febrows exists then
      set var febsar to .febsar + .febrows
   endif
   compute marrows as count sardate from sarsum1 where +
   sardate >= .mardate1 and sardate <= .mardate2
   if marrows exists then
      set var marsar to .marsar + .marrows
   endif
   compute aprrows as count sardate from sarsum1 where +
   sardate >= .aprdate1 and sardate <= .aprdate2
   if aprrows exists then
      set var aprsar to .aprsar + .aprrows
   endif
   compute mayrows as count sardate from sarsum1 where +
   sardate >= .maydate1 and sardate <= .maydate2
   if mayrows exists then
      set var maysar to .maysar + .mayrows
   endif
   compute junrows as count sardate from sarsum1 where +
   sardate >= .jundate1 and sardate <= .jundate2
   if junrows exists then
      set var junsar to .junsar + .junrows
   endif
   compute julrows as count sardate from sarsum1 where +
   sardate >= .juldate1 and sardate <= .juldate2
   if julrows exists then
   set var julsar to .julsar + .julrows
   endif
   compute augrows as count sardate from sarsum1 where +
   sardate >= .augdate1 and sardate <= .augdate2
   if augrows exists then
   set var augsar to .augsar + .augrows
endif
compute seprows as count sardate from sarsuml where +
sardate >= .sepdate1 and sardate <= .sepdate2
if seprows exists then
   set var sepsar to .sepsar + .seprows
endif
compute octrows as count sardate from sarsuml where +
sardate >= .octdate1 and sardate <= .octdate2
if octrows exists then
   set var octsar to .octsar + .octrows
endif
compute novrows as count sardate from sarsuml where +
sardate >= .novdate1 and sardate <= .novdate2
if novrows exists then
   set var novsar to .novsar + .novrows
endif
compute decrows as count sardate from sarsuml where +
sardate >= .decdate1 and sardate <= .decdate2
if decrows exists then
   set var decsar to .decsar + .decrows
endif
set var yrtest1 to .saryear / 4
set var yrtest2 to .yrtest1 x 4
set var yrtest to .saryear - .yrtest2
if yrtest = 0 then
   set var jandate1 to .jandate1 + 366
   set var jandate2 to .jandate2 + 366
   set var febdate1 to .febdate1 + 366
   set var febdate2 to .febdate2 + 366
   set var mardate1 to .mardate1 + 366
   set var mardate2 to .mardate2 + 366
   set var aprdate1 to .aprdate1 + 366
   set var aprdate2 to .aprdate2 + 366
   set var maydate1 to .maydate1 + 366
   set var maydate2 to .maydate2 + 366
   set var jundate1 to .jundate1 + 366
   set var jundate2 to .jundate2 + 366
   set var juldate1 to .juldate1 + 366
   set var juldate2 to .juldate2 + 366
   set var augdate1 to .augdate1 + 366
   set var augdate2 to .augdate2 + 366
   set var sepdate1 to .sepdate1 + 366
   set var sepdate2 to .sepdate2 + 366
   set var octdate1 to .octdate1 + 366
   set var octdate2 to .octdate2 + 366
   set var novdate1 to .novdate1 + 366
   set var novdate2 to .novdate2 + 366
   set var decdate1 to .decdate1 + 366
   set var decdate2 to .decdate2 + 366
else
   set var jandate1 to .jandate1 + 365
   set var jandate2 to .jandate2 + 365
   set var febdate1 to .febdate1 + 365
   set var febdate2 to .febdate2 + 365
   set var mardate1 to .mardate1 + 365
   set var mardate2 to .mardate2 + 365
   set var aprdate1 to .aprdate1 + 365
   set var aprdate2 to .aprdate2 + 365
   set var maydate1 to .maydate1 + 365
   set var maydate2 to .maydate2 + 365
   set var jundate1 to .jundate1 + 365
   set var jundate2 to .jundate2 + 365
   set var juldate1 to .juldate1 + 365
   set var juldate2 to .juldate2 + 365
   set var augdate1 to .augdate1 + 365
   set var augdate2 to .augdate2 + 365
   set var sepdate1 to .sepdate1 + 365
   set var sepdate2 to .sepdate2 + 365
   set var octdate1 to .octdate1 + 365
```

```
      set var octdate2 to .octdate2 + 365
      set var novdate1 to .novdate1 + 365
      set var novdate2 to .novdate2 + 365
      set var decdate1 to .decdate1 + 365
      set var decdate2 to .decdate2 + 365
   endif
   set var count to .count + 1
endwhile
newpage
compute totsars as rows from sarsum1
set var pctjan real
set var pctfeb real
set var pctmar real
set var pctapr real
set var pctmay real
set var pctjun real
set var pctjul real
set var pctaug real
set var pctsep real
set var pctoct real
set var pctnov real
set var pctdec real
     *(calculating percentages based on total SAR cases)
set var pctjan to .jansar / .totsars
set var pctfeb to .febsar / .totsars
set var pctmar to .marsar / .totsars
set var pctapr to .aprsar / .totsars
set var pctmay to .maysar / .totsars
set var pctjun to .junsar / .totsars
set var pctjul to .julsar / .totsars
set var pctaug to .augsar / .totsars
set var pctsep to .sepsar / .totsars
set var pctoct to .octsar / .totsars
set var pctnov to .novsar / .totsars
set var pctdec to .decsar / .totsars
newpage
write "Preparing report...."
if numrows fails or numrows = 0 then
   newpage
   write "No records exist in the database matching your selection +
criteria" at 10 1
   write "Please press any key to continue" at 16 10
   pause
else
   output printer with sardates.rpt
   newpage
   print sardates
   output screen
endif
remove sarsum
remove sarstat
newpage
clear numrows
project sarsum from sar using folderno sarday sardate where +
sarday exists and sardate exists
compute numrows as rows from sarsum
compute monday as count sarday from sarsum where sarday eq mon
compute tuesday as count sarday from sarsum where sarday eq tue
compute wedsday as count sarday from sarsum where sarday eq wed
compute thursday as count sarday from sarsum where sarday eq thu
compute friday as count sarday from sarsum where sarday eq fri
compute saturday as count sarday from sarsum where sarday eq sat
compute sunday as count sarday from sarsum where sarday eq sun
compute totdays as count sarday from sarsum where sarday exists
set var pctmon real
set var pcttue real
set var pctwed real
set var pctthu real
set var pctfri real
set var pctsat real
set var pctsun real
set var pctmon to .monday / .totdays
set var pcttue to .tuesday / .totdays
set var pctwed to .wedsday / .totdays
set var pctthu to .thursday / .totdays
set var pctfri to .friday / .totdays
set var pctsat to .saturday / .totdays
set var pctsun to .sunday / .totdays
write "Processing of statistical information completed." at 10 1
write "Now preparing report..........." at 11 1
remove temp
define
attributes
tmp text 2
relations
temp with tmp
end
load temp
   "XX"
end
if numrows fails or numrows = 0 then
   newpage
   write "No records exist in the database matching your selection +
criteria" at 10 1
   write "Please press any key to return to menu" at 16 10
   pause
else
   output printer with sardays.rpt
   newpage
   print sardays
   newpage
   output screen
endif
clear all variables
remove temp
remove sarstat
newpage
set escape off
return
*( end )


$COMMAND
DAYNIGHT
*(***********************************************************************

    Program:           Daytime vs. Nighttime Frequency Analysis
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program performs a statistical analysis
                       of SAR cases and Boardings to determine
                       percentage of cases which occur in the
                       daytime vs those that occur at night during
                       a specified time period.
    Tables Used:       sar, boarding
    Temp Tables:       sarsum, bosum, temp
    Reports Used:      daynight
    Forms Used:        none

************************************************************************)

set null " "
set escape on
set messages off
set error messages off
clear all variables
remove sarsum
newpage
```

```
write "This program performs a statistical analysis of Search &" at 2 1
write "Rescue cases and Boardings to determine the percentage of " +
at 3 1
write "cases which occurred during the day and at night during a " +
at 4 1
write "specified time period. " at 5 1
set var strtdate date
set var enddate date
set var datediff to 0
set error messages on
while datediff <= 0 then
   write "Please enter starting date in the form MM/DD/YYYY:" at 8 1
   fillin strtdate using " " at 8 51
   write "Please enter ending date in the form MM/DD/YYYY:" at 10 1
   fillin enddate using " " at 10 51
   set var datediff to .enddate - .strtdate
endwhile
set error messages off
project sarsum from sar using ssunrise ssunset sartime sardate where +
sardate >= .strtdate and sardate <= .enddate and ssunrise exists and +
sartime exists
compute numrows as rows from sarsum
compute daysar as count sartime from sarsum where sartime gea +
ssunrise and sartime lta ssunset
compute nitesar as count sartime from sarsum where sartime lta +
ssunrise or sartime gea ssunset
set var totsars to .daysar + .nitesar
set var pctdays real
set var pctnites real
set var pctdays to .daysar / .totsars
set var pctnites to .nitesar / .totsars
set var pctdays to .pctdays x 100.0
set var pctnites to .pctnites x 100.0
remove sarsum
remove bosum
newpage
project bosum from boarding using bsunset bsunrise botime bodate where +
bodate >= .strtdate and bodate <= .enddate and bsunrise exists and +
botime exists
compute numrows1 as rows from bosum
compute daybrd as count botime from bosum where botime gea bsunrise +
and botime lta bsunset
compute nitebrd as count botime from bosum where botime lta bsunrise +
or botime gea bsunset
set var totbrds to .daybrd + .nitebrd
set var pctdayb real
set var pctniteb real
set var pctdayb to .daybrd / .totbrds
set var pctniteb to .nitebrd / .totbrds
set var pctdayb to .pctdayb x 100.0
set var pctniteb to .pctniteb x 100.0
remove bosum
remove temp
newpage
define
attributes
tmp text 2
relations
temp with tmp
end
load temp
   "XX"
end
newpage
write "Preparing report....."
if numrows fails or numrows = 0 then
   if numrows1 fails or numrows1 = 0 then
      newpage
      write "No records exist matching your selection criteria" +
      at 10 1
      write "Please press any key to return to menu" at 16 10
      pause
   else
      output printer with daynight.rpt
      newpage
      print daynight
      newpage
      output screen
   endif
else
   output printer
   newpage
   print daynight
   newpage
   output screen
endif
remove sarsum
remove bosum
remove temp
newpage
return
*( end )


$COMMAND
SARDUMP
*(*************************************************************************

    Program:              SAR Summary Listing Program
    Author:               J D Allen and H R Lang
    Date Written:         August 1987
    Description:          This program performs a dump of the table SAR
                          of those.SAR cases that occurred during a
                          specified time period.  The listing is sorted
                          by date prior to being printed.  .
    Tables Used:          sar
    Temp Tables:          sarsum
    Reports Used:         sar
    Forms Used:           none

*************************************************************************)

set null " "
set escape on
set messages off
set error messages off
newpage
clear all variables
write "This program will dump all information on SAR cases " at 3 1
write "which are selected based on SAR date.  If you wish to print" +
at 4 1
write "out the entire database of SAR cases, please enter the date " +
at 5 1
write "[01/01/1900] for the starting date and todays date for the " +
at 6 1
write "ending date............................................... " +
at 7 1
set var strtdate date
set var enddate date
set var datediff to 0
set error messages on
while datediff <= 0 then
   write "Please enter starting date in the form MM/DD/YYYY:" at 9 1
   fillin strtdate using " " at 9 53
   write "Please enter ending date in the form MM/DD/YYYY:" at 11 1
   fillin enddate using " " at 11 53
   set var datediff to .enddate - .strtdate
endwhile
```

```
set error messages off
remove sarsum
newpage
project sarsum from sar using all sorted by sardate where +
sardate >= .strtdate and sardate <= .enddate
compute numrows as rows from sarsum
newpage
write "Now preparing report............" at 11 1
if numrows fails or numrows = 0 then
   newpage
   write "No records exist matching your selection criteria" +
   at 10 1
   write "Please press any key to return to menu" at 16 10
   pause
else
   output printer with sar.rpt
   newpage
   print sar
   newpage
   output screen
endif
remove sarsum
newpage
clear all variables
return
*( end )


$COMMAND
IMMEDSAR
*(**********************************************************************

    Program:           Immediate SAR Frequency Program
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program performs a statistical analysis
                       of SAR cases to determine percentage of cases
                       which are of immediate nature during a
                       specified time period.
    Tables Used:       sar
    Temp Tables:       sarsum, temp
    Reports Used:      immedsar
    Forms Used:        none

***********************************************************************)

set null " "
set escape on
set messages off
set error messages off
clear all variables
remove sarsum
newpage
write "This program performs a statistical analysis of Search &" at 2 1
write "Rescue cases to determine the percentage of cases which " at 3 1
write "are considered to be of an immediate nature or of " at 4 1
write "immediate peril to the vessel during a specified time period" +
at 5 1
set var strtdate date
set var enddate date
set var datediff to 0
set error messages on
while datediff <= 0 then
   write "Please enter starting date in the form MM/DD/YYYY:" at 8 1
   fillin strtdate using " " at 8 51
   write "Please enter ending date in the form MM/DD/YYYY:" at 10 1
   fillin enddate using " " at 10 51
   set var datediff to .enddate - .strtdate
```

```
endwhile
set error messages off
remove sarsum
newpage
project sarsum from sar using folderno immedsar sardate where sardate +
>= .strtdate and sardate <= .enddate
compute numrows as rows from sarsum
compute totsars as count folderno from sarsum where folderno exists
compute totimmed as count immedsar from sarsum where immedsar = Y
set var pctimmed real
set var pctnotim real
set var pctimmed to .totimmed / .totsars
set var notimmed to .totsars - .totimmed
set var pctnotim to .notimmed / .totsars
remove temp
*(defining temporary relation for printing)
define
attributes
tmp text 2
relations
temp with tmp
end
*(loading temporary relation with data)
load temp
   "XX"
end
newpage
write "Now preparing Immediate SAR case report....." at 10 1
if numrows fails or numrows = 0 then
   newpage
   write "No records exist matching your selection criteria" +
   at 10 1
   write "Please press any key to return to menu" at 16 10
   pause
else
   output printer with immedsar.rpt
   newpage
   print immedsar
   output screen
endif
remove sarsum
remove temp
newpage
clear all variables
return
*( end )
```

APPENDIX M

VSLRPTS SOURCE CODE

```
$COMMAND
VSL_MAIN
*(**********************************************************************

                     US COAST GUARD GROUP MONTEREY
          SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                     INTELLIGENCE SYSTEM (SARELTIS)

              VESSEL REPORTS MENU and APPLICATION PROGRAMS


                 Author:    Jon D. Allen, LT, USCG
                            Heidi R. Lang, LT, USN
                            AUGUST 1987


**********************************************************************)

set error messages off
set messages off
set escape off
clear all variables
newpage
set var pick8 integer
choose pick8 from vsl_menu in vslrpts.apx
if pick8 eq 0 then
   quit to restvsl.cmd
endif
if pick8 eq 1 then
   run VESSEL in vslrpts.apx
   quit to restvsl.cmd
endif
if pick8 eq 2 then
   run VSLOWNER in vslrpts.apx
   quit to restvsl.cmd
endif
if pick8 eq 3 then
   run VSLOPER in vslrpts.apx
   quit to restvsl.cmd
endif
if pick8 eq 4 then
   quit to restrpts.cmd
endif
quit to restvsl.cmd
return
$menu
vsl_menu
column SARELTIS --- VESSEL REPORTS
VESSEL SUMMARY LISTING
VESSEL --- OWNERS OF SPECIFIED VESSELS
VEESEL --- OPERATORS OF SPECIFIED VESSELS
RETURN TO REPORTS MENU
*( end )


$COMMAND
VESSEL
*(**********************************************************************

    Program:          Vessel Summary Listing Program
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program performs a dump of the VESSEL
                       table.  Listing entries are selected by
                       vessel type, vessel use, and by an
                       alphabetic search.
    Tables Used:       vessel, boattype, boatuse
    Temp Tables:       vslsum, vslsum1, vslsum2
    Reports Used:      vessel
    Forms Used:        none

*********************************************************************)

set null " "
set error messages off
set messages off
set escape on
newpage
clear all variables
write "This program will dump all information on vessels "
write "which are selected based on an alphabetic search,  vessel  "
write "type search,  and a vessel use search.                     "
write "The following is a list a legal codes for vessel type (they "
write "can be modified by changing/adding codes to the BOATTYPE table):"
write " "
select all from boattype sorted by vtype
set var legaltyp text
set var typchoic text
set var legaltyp to no
while legaltyp eq no then
   set error messages on
   fillin typchoic using "Please enter in vessel type or [XX] for all +
types: "
   set error messages off
   if typchoice eq XX then
      set var anytype to yes
      set var legaltyp to yes
      break
   else
      set var anytype to no
   endif
   set pointer #1 n1 for boattype
   while n1 eq 0 then
      set var typetest to vtype in #1
      if typchoic eq .typetest then
         set var legaltyp to yes
         break
      endif
      next #1 n1
   endwhile
endwhile
newpage
set var usechoic text
set var legaluse to no
write "The following is a list of legal codes for vessel uses.  These "
write "be modified by adding/changing the values in the BOATUSE table."
write "   "
select all from boatuse sorted by vuse
set var legaluse text
set var usechoic text
set var legaluse to no
while legaluse eq no then
   set error messages on
   fillin usechoic using "Please enter vessel use or XX for all types: "
   set error messages off
   if usechoic eq XX then
      set var legaluse to yes
      set var anyuse to yes
      break
   else
      set var anyuse to no
   endif
   set pointer #1 n1 for boatuse
   while n1 eq 0 then
      set var usetest to vuse in #1
      if usechoic eq .usetest then
         set var legaluse to yes
         break
      endif
      next #1 n1
   endwhile
endwhile
newpage
write "Finally,  you will be asked to enter a single letter for both +
the"
write "start and the end of an alphabetic grouping.  For example,  if "
write "you enter an [e] for the start and an [i] for the end,  the "
write "program will only list those vessels whose names begin with the"
write "letters [e] through [i] inclusively.  To select the entire +
vessel"
write "database,  please enter [a] and [z] in their corresponding +
fields"
set var strtltr text
set var endltr text
set var ltrdiff to 0
set error messages on
while ltrdiff = 0 then
   fillin strtltr using "Please enter starting letter:  " at 14 1
   fillin endltr using "Please enter ending letter:    " at 16 1
   if endltr lt .strtltr then
      set var ltrdiff to 0
   else
      break
   endif
endwhile
set error messages off
newpage
write "You now have the option to select how you would like the +
report" at 3 1
write "to appear.  It can be sorted by vessel name,  vessel HIN,  " +
at 4 1
write "vessel type,  or vessel use." at 5 1
set var choice text
set var legalch to no
while legalch eq no then
   write "1   Sorted by vessel name" at 8 1
   write "2   Sorted by vessel HIN" at 9 1
   write "3   Sorted by vessel type" at 10 1
   write "4   Sorted by vessel use" at 11 1
   set error messages on
   fillin choice using "Please enter your choice:  " at 13 1
   set error messages off
   if choice eq 1 or choice eq 2 or choice eq 3 or choice eq 4 then
      set var legalch to yes
      break
   else
      set var legalch to no
   endif
endwhile
remove vslsum1
remove vslsum2
remove vslsum
     *(produces vessel listing sorted by vessel name)
if choice eq 1 then
   project vslsum1 from vessel using all sorted by vslname where +
   vslname ge .strtltr and vslname le .endltr
   if anytype eq yes then
      project vslsum2 from vslsum1 using all
   else
```

```
      project vslsum2 from vslsum1 using all where vsltype eq .typchoic
   endif
   if anyuse eq yes then
      project vslsum from vslsum2 using all
   else
      project vslsum from vslsum2 using all where vsluse eq .usechoic
   endif
endif
      *(produces vessel listing sorted by vessel HIN)
if choice eq 2 then
   project vslsum1 from vessel using all sorted by vslhin where +
   vslname ge .strtltr and vslname le .endltr
   if anytype eq yes then
      project vslsum2 from vslsum1 using all
   else
      project vslsum2 from vslsum1 using all where vsltype eq .typchoic
   endif
   if anyuse eq yes then
      project vslsum from vslsum2 using all
   else
      project vslsum from vslsum2 using all where vsluse eq .usechoic
   endif
endif
     *(produces vessel listing sorted by vessel type)
if choice eq 3 then
   project vslsum1 from vessel using all where vslname ge .strtltr and +
   vslname le .endltr
   if anytype eq yes then
      project vslsum2 from vslsum1 using all sorted by vsltype
   else
      project vslsum2 from vslsum1 using all sorted by vsltype where +
      vsltype eq .typchoice
   endif
   if anyuse eq yes then
      project vslsum from vslsum2 using all
   else
      project vslsum from vslsum2 using all where vsluse eq .usechoic
   endif
endif
      *(produces vessel listing sorted by vessel use)
if choice eq 4 then
   project vslsum1 from vessel using all where vslname ge .strtltr and +
   vslname le .endltr
   if anytype eq yes then
      project vslsum2 from vslsum1 using all
   else
      project vslsum2 from vslsum1 using all where vsltype eq .typchoic
   endif
   if anyuse eq yes then
      project vslsum from vslsum2 using all sorted by vsluse
   else
      project vslsum from vslsum2 using all sorted by vsluse where +
      vsluse eq .usechoic
   endif
endif
compute numrows as rows from vslsum
if numrows fails or numrows = 0 then
   write "No records match selection criteria" at 10 10
   write "Please press any key to continue" at 16 10
   pause
else
   output printer with vessel.rpt
   newpage
   print vessel
   output screen
endif
remove vslsum
remove vslsum1
remove vslsum2
newpage
```

```
clear all variables
return
*( end )


$COMMAND
VSLOWNER
*(**********************************************************************

    Program:            Vessel - Owner Dump Program
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This program performs a listing sorted by
                        vessel name of those vessels which are
                        owned by a specified group of owners.  All
                        of the vessels in the VESSEL table may be
                        selected or only a group specified in an
                        alphabetic search.
    Tables Used:        owvsl, owners, vessel
    Temp Tables:        ownvsls, ownvsls1, ownvsls2
    Reports Used:       vslowner
    Forms Used:         none

**********************************************************************)

set null " "
set error messages off
set messages off
set escape on
newpage
clear all variables
write "This program will prepare a listing of owners which" at 3 1
write "own by a specified group of vessels.  This report can" at 4 1
write "provide a listing of all owners and their vessels by" at 5 1
write "responding [y] to the following question......" at 6 1
write "Do you wish a complete listing of all owners and their" at 9 1
write "vessels contained in the system?" at 10 1
set var allrecs text
set error messages on
fillin allrecs using " " at 10 40
set error messages off
remove ownvsls
remove ownvsls1
remove ownvsls2
newpage
if allrecs contains "y" then
   join owners using ownid with owvsl using idowner forming ownvsls1
   join ownvsls1 using vhinown with vessel using vslhin forming +
   ownvsls2
   project ownvsls from ownvsls2 using ownlname ownfname ownmi owndob +
   ownaddr owncity ownstate ownzip owntelno vslname vslhin sorted by +
   vslname
else
   newpage
   set var strtltr text
   set var endltr text
   set var ltrdiff to 0
   set error messages on
   while ltrdiff = 0 then
      write "Enter starting letter(s) to start search:" at 5 1
      fillin strtltr using " " at 5 45
      write "Enter ending letter(s) to end search:" at 7 1
      fillin endltr using " " at 7 45
      if endltr lt .strtltr then
         set var ltrdiff to 0
      else
         break
      endif
   endwhile
   set error messages off
   join owners using ownid with owvsl using idowner forming ownvsls1
   join ownvsls1 using vhinown with vessel using vslhin forming +
   ownvsls2
   project ownvsls from ownvsls2 using ownlname ownfname ownmi owndob +
   ownaddr owncity ownstate ownzip owntelno vslname vslhin sorted by +
   vslname where vslname ge .strtltr and vslname le .endltr
endif
remove ownvsls1
remove ownvsls2
newpage
compute numrows as rows from ownvsls
if numrows fails or numrows = 0 then
   write "No records match selection criteria" at 10 10
   write "Please press any key to continue" at 16 10
   pause
else
   output printer with vslowner.rpt
   newpage
   print vslowner
   output screen
endif
remove ownvsls
newpage
clear all variables
return
*( end )


$COMMAND
VSLOPER
*(**************************************************************************

    Program:            Vessel - Operator Dump Program
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This program performs a listing sorted by
                        vessel name of those vessels which are
                        operated by a specified group of operators.
                        The entire list of vessels may be printed
                        or a specified group based on an alphabetic
                        search.
    Tables Used:        opvsl, operator, vessel
    Temp Tables:        opervsls, opvsls1, opvsls2
    Reports Used:       vsloper
    Forms Used:         none

**************************************************************************)

set null " "
set error messages off
set messages off
set escape on
newpage
clear all variables
write "This program will prepare a listing of operators which" at 3 1
write "operate a specified group of vessels.  This report can" at 4 1
write "provide a listing of all operators and their vessels by" at 5 1
write "responding [y] to the following question......" at 6 1
write "Do you wish a complete listing of all operators and their" +
at 9 1
write "vessels contained in the system?" at 10 1
set var allrecs text
set error messages on
fillin allrecs using " " at 10 40
set error messages off
remove opervsls
remove opvsls1
remove opvsls2
newpage
if allrecs contains "y" then
   join operator using opid with opvsl using idoper forming opvsls1
   join opvsls1 using vhinoper with vessel using vslhin forming opvsls2
   project opervsls from opvsls2 using oplname opfname opmi opdob +
   opaddr opcity opstate opzip optelno vslname vslhin sorted by vslname
else
   newpage
   set var strtltr text
   set var endltr text
   set var ltrdiff to 0
   set error messages on
   while ltrdiff = 0 then
      write "Enter starting letter(s) to start search:" at 5 1
      fillin strtltr using " " at 5 45
      write "Enter ending letter(s) to end search:" at 7 1
      fillin endltr using " " at 7 45
      if endltr lt .strtltr then
         set var ltrdiff to 0
      else
         break
      endif
   endwhile
   set error messages off
   join operator using opid with opvsl using idoper forming opvsls1
   join opvsls1 using vhinoper with vessel using vslhin forming opvsls2
   project opervsls from opvsls2 using oplname opfname opmi opdob +
   opaddr opcity opzip opstate optelno vslname vslhin sorted by +
   vslname where vslname ge .strtltr and vslname le .endltr
endif
compute numrows as rows from opervsls
remove opvsls1
remove opvsls2
newpage
if numrows fails or numrows = 0 then
   write "No records match selection criteria" at 10 10
   write "Please press any key to continue" at 16 10
   pause
else
   output printer with vsloper.rpt
   newpage
   print vsloper
   output screen
endif
remove opvsls
newpage
clear all variables
return
*( end )
```

APPENDIX N
PRSRPTS SOURCE CODE

```
$COMMAND
PRS_MAIN
*(***********************************************************************

                     US COAST GUARD GROUP MONTEREY
         SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                     INTELLIGENCE SYSTEM (SARELTIS)

            PERSONNEL REPORTS MENU and APPLICATION PROGRAMS

                 Author:     Jon D. Allen, LT, USCG
                             Heidi R. Lang, LT, USN
                             AUGUST 1987

***********************************************************************)

set error messages off
set messages off
set escape off
clear all variables
newpage
set var pick9 integer
choose pick9 from prs_menu in prsrpts.apx
if pick9 eq 0 then
   quit to restprs.cmd
endif
if pick9 eq 1 then
   run OWNERS in prsrpts.apx
   quit to restprs.cmd
endif
if pick9 eq 2 then
   run OWNVSL in prsrpts.apx
   quit to restprs.cmd
endif
if pick9 eq 3 then
   run OPERATOR in prsrpts.apx
   quit to restprs.cmd
endif
if pick9 eq 4 then
   run OPERVSLS in prsrpts.apx
   quit to restprs.cmd
endif
if pick9 eq 5 then
   quit to restrpts.cmd
endif
quit to restprs.cmd
return
$menu
prs_menu
column SARELTIS --- PERSONNEL REPORTS
OWNERS OF VESSELS SUMMARY LISTING
OWNER --- VESSELS WHICH ARE OWNED BY SPECIFIED OWNERS
OPERATORS OF VESSELS SUMMARY LISTING
OPERATORS --- VESSELS WHICH ARE OPERATED BY SPECIFIED OPERATORS
RETURN TO REPORTS MENU
*( end )
```

```
$COMMAND
OWNERS
*(***********************************************************************

    Program:            Vessel Owners Table Dump
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This program prints a sorted listing of
                        vessel owners contained in the database
                        based on the owner's last name and date
                        of birth.
    Tables Used:        owners
    Temp Tables:        ownsum1, ownsum2
    Reports Used:       owners
    Forms Used:         none

***********************************************************************)

set null " "
set error messages off
set messages off
set escape on
newpage
clear all variables
write "This program will dump all information on owners in the +
database" at 3 1
write "which are selected based on date of birth.  If you wish to +
print" at 4 1
write "out the entire list of owners,  enter the date [01/01/1900] +
for " at 5 1
write "the starting date and todays date for the ending date." at 6 1
set var datediff to 0
set error messages on
while datediff <= 0 then
   write "Please enter starting date in the form MM/DD/YYYY:" at 8 1
   fillin strtdate using " " at 8 53
   write "Please enter ending date in the form MM/DD/YYYY:" at 10 1
   fillin enddate using " " at 10 53
   set var datediff to .enddate - .strtdate
endwhile
set error messages off
newpage
write "Since this listing can be very long, you have the option of +
only" at 3 1
write "printing a selected alphabetical grouping of owners. You will +
be" at 4 1
write "asked to enter a single letter for the both the start and end +
of" at 5 1
write "an alphabetic grouping.  For example,  if you enter in an [e] +
for" at 6 1
write "start and an [i] for the end,  the program will provide a +
listing" at 7 1
write "of only those owners whose last names begin with the letters " +
at 8 1
write "[e] through [i].  To select the entire owner database,  +
please" at 9 1
write "enter [a] and [z] in their corresponding fields..........." +
at 10 1
set var strtltr text
set var endltr text
set var ltrdiff to 0
set error messages on
while ltrdiff = 0 then
   fillin strtltr using "Please enter starting letter:  " at 14 1
   fillin endltr using "Please enter ending letter:     " at 16 1
   if endltr lt .strtltr then
      set var ltrdiff to 0
   else
      break
   endif
endwhile
set error messages off
remove ownsum1
remove ownsum2
newpage
project ownsum1 from owners using all sorted by ownlname where +
owndob >= .strtdate and owndob <= .enddate
project ownsum2 from ownsum1 using all where ownlname ge .strtltr and +
ownlname le .endltr
compute numrows as rows from ownsum2
newpage
write "Processing of statistical information completed." at 10 1
write "Now preparing report............" at 11 1
if numrows fails or numrows = 0 then
   write "No records match selection criteria" at 10 10
   write "Please press any key to continue" at 16 10
   pause
else
   output printer with owners.rpt
   newpage
   print owners
   newpage
   output screen
endif
remove ownsum1
remove ownsum2
newpage
clear all variables
return
*( end )


$COMMAND
OPERATOR
*(**********************************************************************

    Program:            Vessel Operator Listing
    Author:             J D Allen and H R Lang
    Date Written:       August 1987
    Description:        This program performs a listing of operators
                        of vessels contained in the table OPERATOR.
                        They are selected by date of birth and by
                        the first letter of the last name.  The
                        listing is sorted alphabetically before
                        being printed.
    Tables Used:        operator
    Temp Tables:        opsum1, opsum2
    Reports Used:       operator
    Forms Used:         none

**********************************************************************)

set null " "
set error messages off
set messages off
set escape on
newpage
clear all variables
write "This program will dump all information on operators " at 3 1
write "which are selected based on date of birth.  If you wish to +
print" at 4 1
write "out the entire list of operators, enter the date [01/01/1900] +
for" at 5 1
write "the starting date and todays date for the ending date." at 6 1
set var strtdate date
set var enddate date
set var datediff to 0
set error messages on
while datediff <= 0 then
   write "Please enter starting date in the form MM/DD/YYYY:" at 8 1
   fillin strtdate using " " at 8 53
   write "Please enter ending date in the form MM/DD/YYYY:" at 10 1
   fillin enddate using " " at 10 53
   set var datediff to .enddate - .strtdate
endwhile
set error messages off
newpage
write "Since this listing can be very long, you have the option of +
only" at 3 1
write "printing a selected grouping of operators.  You will be" at 4 1
write "asked to enter a single letter for the both the start and end +
of " at 5 1
write "an alphabetic grouping.  For example,  if you enter in an [e] +
for" at 6 1
write "start and an [i] for the end,  the program will provide a +
listing" at 7 1
write "of only those operators whose last names begin with the +
letters " at 8 1
write "[e] through [i].  To select the entire operator database,  +
please" at 9 1
write "enter [a] and [z] in their corresponding fields..........." +
at 10 1
set var strtltr text
set var endltr text
set var ltrdiff to 0
set error messages on
while ltrdiff = 0 then
   fillin strtltr using "Please enter starting letter:  " at 14 1
   fillin endltr using "Please enter ending letter:    " at 16 1
   if endltr lt .strtltr then
      set var ltrdiff to 0
   else
      break
   endif
endwhile
set error messages off
remove opsum1
remove opsum2
newpage
project opsum1 from operator using all sorted by oplname where +
opdob >= .strtdate and opdob <= .enddate
project opsum2 from opsum1 using all where oplname ge .strtltr and +
oplname le .endltr
compute numrows as rows from opsum2
write "Processing of statistical information completed." at 10 1
write "Now preparing report............" at 11 1
if numrows fails or numrows = 0 then
   write "No records match selection criteria" at 10 10
   write "Please press any key to continue" at 16 10
   pause
else
   output printer with operator.rpt
   newpage
   print operator
   output screen
endif
remove opsum1
remove opsum2
newpage
clear all variables
return
*( end )


$COMMAND
OPERVSLS
```

```
*(*********************************************************************

    Program:           Operator - Vessel Program
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program performs a listing sorted by
                       operator last name of those vessels which
                       are used by specified operators.  Operators
                       may be selected by an alphabetic search or
                       may be selected in their entirety.
    Tables Used:       opvsl, operator, vessel
    Temp Tables:       opervsls, opvsls1, opvsls2
    Reports Used:      opervsls
    Forms Used:        none

 *********************************************************************)
set null " "
set error messages off
set messages off
set escape on
newpage
clear all variables
write "This program will prepare a listing of vessels which are" at 3 1
write "operated by a selected group of operators.  This report can" +
at 4 1
write "provide a listing of all operators and their vessels by" at 5 1
write "responding [y] to the following question......" at 6 1
write "Do you wish a complete listing of all operators and their" +
at 9 1
write "vessels contained in the system?" at 10 1
set var allrecs text
set error messages on
fillin allrecs using " " at 10 40
set error messages off
remove opervsls
remove opvsls1
remove opvsls2
if allrecs = "y" then
   join operator using opid with opvsl using idoper forming opvsls1
   join opvsls1 using vhinoper with vessel using vslhin forming opvsls2
   project opervsls from opvsls2 using oplname opfname opmi opdob +
   opaddr opcity opstate opzip optelno vslname vslhin sorted by oplname
else
   newpage
   set var strtltr text
   set var endltr text
   set var ltrdiff to 0
   set error messages on
   while ltrdiff = 0 then
      write "Enter starting letter(s) to start search:" at 5 1
      fillin strtltr using " " at 5 45
      write "Enter ending letter(s) to end search:" at 7 1
      fillin endltr using " " at 7 45
      if endltr lt .strtltr then
         set var ltrdiff to 0
      else
         break
      endif
   endwhile
   set error messages off
   join operator using opid with opvsl using idoper forming opvsls1
   join opvsls1 using vhinoper with vessel using vslhin forming opvsls2
   project opervsls from opvsls2 using oplname opfname opmi opdob +
   opaddr opcity opstate opzip optelno vslname vslhin sorted by +
   oplname where oplname ge .strtltr and oplname le .endltr
endif
compute numrows as rows from opervsls
remove opvsls1
remove opvsls2
newpage
if numrows fails or numrows = 0 then
   write "No records match selection criteria" at 10 10
   write "Please press any key to continue" at 16 10
   pause
else
   output printer with opervsls.rpt
   newpage
   print opervsls
   output screen
endif
remove opervsls
newpage
clear all variables
return
*( end )


$COMMAND
OWNVSL
*(***********************************************************************

     Program:             Owner - Vessel Dump Program
     Author:              J D Allen and H R Lang
     Date Written:        August 1987
     Description:         This program performs a listing sorted by
                          owner last name of those owners which own
                          specific vessels.  Owners may be selected
                          in their entirety or may be selected by
                          an alphabetic search.
     Tables Used:         owvsl, owners, vessel
     Temp Tables:         ownvsls, ownvsls1, ownvsls2
     Reports Used:        ownvsls
     Forms Used:          none

***********************************************************************)

set null " "
set error messages off
set messages off
set escape on
newpage
clear all variables
write "This program will prepare a listing of vessels which are" at 3 1
write "owned by a selected group of owners.  This report can" at 4 1
write "provide a listing of all owners and their vessels by" at 5 1
write "responding [y] to the following question......" at 6 1
write "Do you wish a complete listing of all owners and their" at 9 1
write "vessels contained in the system?" at 10 1
set var allrecs text
set error messages on
fillin allrecs using " " at 10 40
set error messages off
remove ownvsls
remove ownvsls1
remove ownvsls2
newpage
if allrecs contains "y" then
   join owners using ownid with owvsl using idowner forming ownvsls1
   join ownvsls1 using vhinown with vessel using vslhin forming +
   ownvsls2
   project ownvsls from ownvsls2 using ownlname ownfname ownmi owndob +
   ownaddr owncity ownstate ownzip owntelno vslname vslhin sorted by +
   ownlname
else
   newpage
   set var strtltr text
   set var endltr text
   set var ltrdiff to 0
   set error messages on
   while ltrdiff = 0 then
      write "Enter starting letter(s) to begin search" at 5 1
      fillin strtltr using " " at 5 45
      write "Enter ending letter(s) to end search:" at 7 1
      fillin endltr using " " at 7 45
      if endltr lt .strtltr then
         set var ltrdiff to 0
      else
         break
      endif
   endwhile
   set error messages off
   join owners using ownid with owvsl using idowner forming ownvsls1
   join ownvsls1 using vninown with vessel using vslhin forming +
   ownvsls2
   project ownvsls from ownvsls2 using ownlname ownfname ownmi owndob +
   ownaddr owncity ownstate ownzip owntelno vslname vslhin sorted by +
   ownlname where ownlname ge .strtltr and ownlname le .endltr
endif
compute numrows as rows from ownvsls
if numrows fails or numrows = 0 then
   write "No records match selection criteria" at 10 10
   write "Please press any key to continue" at 16 10
   pause
else
   output printer with ownvsls.rpt
   newpage
   print ownvsls
   output screen
endif
remove ownvsls
remove ownvsls1
remove ownvsls2
newpage
clear all variables
return
*( end )
```

APPENDIX O

SQUERY SOURCE CODE

```
$COMMAND
QRYMAIN
*(**********************************************************************

                     US COAST GUARD GROUP MONTEREY
         SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                     INTELLIGENCE SYSTEM (SARELTIS)

                  QUERY MENU and APPLICATION PROGRAMS

                 Author:    Jon D. Allen, LT, USCG
                            Heidi R. Lang, LT, USN
                            AUGUST 1987

***********************************************************************)

set messages off
set escape off
newpage
clear all variables
set var pick10 integer
newpage
choose pick10 from qry_menu in squery.apx
if pick10 eq 0 then
   quit to restqry.cmd
endif
if pick10 eq 1 then
   run quer_vsl in squery.apx
   quit to restqry.cmd
endif
if pick10 eq 2 then
   run quer_per in squery.apx
   quit to restqry.cmd
endif
if pick10 eq 3 then
   quit to restmain.cmd
endif
quit to restqry.cmd
return
$menu
qry_menu
column SARELTIS --- QUERY MENU
VESSEL QUERY
PERSONNEL QUERY
RETURN TO MAIN MENU
*( end )


$COMMAND
QUER_VSL
*(**********************************************************************

    Program:              Query Vessel
    Author:               H R Lang and J D Allen
    Date Written:         August 1987
    Description:          This program searches database for all
                          information concerning a particular vessel.
    Tables Used:          vessel, sar, boarding, owner, operator,
                          owvsl, opvsl, vslintel, perintel
    Temp Tables:          temp, vslint, perint, bohis, sarhis
    Reports Used:         vslqprn, boqprn, sarqprn, owqprn, opqprn,
                          pintqprn, vintqprn
    Forms Used:           vslquery, brdquery, sarquery, ownquery,
                          opquery, pintquer, vintquer

************************************************************************)

set messages off
set escape off
newpage
write "Program loading in progress.  Please stand by ..........." +
at 10 10
clear all variables
set null " "
set var more to Y
remove temp  *(relation for print purposes only)
define
attributes
tmp text 2
relations
temp with tmp
end
load temp
  "XX"
end
while more eq Y then  *(main loop)
  set var vvslname text
  set var choice to 0
  while choice lt 1 or choice gt 6 then
    newpage
    write "Search Option Menu" at 5 15
    write "------------------" at 6 15
    write "1.  Search by HIN" at 8 15
    write "2.  Search by Vessel Name" at 9 15
    write "3.  Search by Vessel Number" at 10 15
    write "4.  Search by Vessel Official Number" at 11 15
    write "5.  Search Only Vessel Intelligence File by Vessel Name" +
at 12 15
    write "6.  Quit" at 13 15
    write "Enter Choice: " at 15 20
    fillin choice using " " at 15 33
  endwhile
  if choice eq 2 then  *(search by vessel name)
    set var chk-name to Y
    while chk-name eq Y then
      newpage
      write "Enter name of desired vessel: " at 10 10
      fillin vvslname using " " at 10 39
      while vvslname exists then
        set pointer #1 e1 for vessel where vslname eq .vvslname
        newpage
        if e1 eq 0 then
          sel vslname vslhin vslmake vslmodel vsltype vsllenft from +
          vessel where vslname eq .vvslname
          write " "
          write "End of data for this vessel name"
          write " "
          write "Press any key to continue"
          pause
        else
          write "This vessel name does not exist in the database" +
at 10 10
          write "Press any key to continue" at 15 15
          pause
          set var choice to 0
        endif
        set var chk-name to N
        break
      endwhile
    endwhile
  endif
  if choice eq 3 then  *(search by vessel number)
    set var chk-no to Y
    while chk-no eq Y then
      newpage
      write "Enter number of desired vessel: " at 10 10
      fillin vvslno using " " at 10 41
      while vvslno exists then
        set pointer #1 e1 for vessel where vslno eq .vvslno
        newpage
        if e1 eq 0 then
          set var vvslname to vslname in #1
          sel vslname vslhin vslmake vslmodel vsltype vsllenft from +
          vessel where vslno eq .vvslno
          write " "
          write "End of data for this vessel number"
          write " "
          write "Press any key to continue"
          pause
        else
          write "This vessel number does not exist in the database" +
at 10 10
          write "Press any key to continue" at 15 15
          pause
          set var choice to 0
        endif
        set var chk-no to N
        break
      endwhile
    endwhile
  endif
  if choice eq 4 then  *(search by vessel official number)
    set var chkoffno to Y
    while chkoffno eq Y then
      newpage
      write "Enter official number of desired vessel: " at 10 10
      fillin voffno using " " at 10 50
      while voffno exists then
        set pointer #1 e1 for vessel where vsloffno eq .voffno
        newpage
        if e1 eq 0 then
          set var vvslname to vslname in #1
          sel vslname vslhin vslmake vslmodel vsltype vsllenft from +
          vessel where vsloffno eq .voffno
          write " "
          write "End of data for this vessel official number"
          write " "
          write "Press any key to continue"
          pause
        else
          write "This official number does not exist in the database" +
at 10 10
          write "Press any key to continue" at 15 15
          pause
          set var choice to 0
        endif
        set var chkoffno to N
        break
      endwhile
    endwhile
  endif
  if choice ne 5 and choice ne 6 and choice ne 0 then  *(search by HIN)
    set var menu to N
    set var chk-hin to Y
    while chk-hin eq Y then
      newpage
      if vvslname exists then
```

```
        set var vlname to .vvslname
        set var nameok to N
        while nameok eq N then
          newpage
          set pointer #3 e3 for vessel where vslname eq .vlname
          while e3 eq 0 then
            newpage
            set var vvslname to vslname in #3
            set var vvslhin to vslhin in #3
            draw hinquery with all at 1
            write "Press [ENTER] to advance to next hin, [PGDN] to +
use displayed data," at 22 5
            write "or [ESC] to return to menu" at 23 5
            edit var vvslhin return esc enter pgdn
            if #return eq enter then
              next #3 e3
            endif
            if #return eq pgdn then
              set var nameok to Y
              break
            endif
            if #return eq esc then
              set var nameok to Y
              set var menu to Y
              break
            endif
          endwhile
        endwhile
        if menu eq Y then
          set var choice to 0
          break
        endif
      else
        set var vvslhin text
        write "Enter HIN of desired vessel: " at 10 10
        write "(or enter  xxx  to go back to menu)" at 12 10
        fillin vvslhin using " " at 10 38
        if vvslhin exists and vvslhin eq xxx then
          set var choice to 0
          break
        endif
      endif
      while vvslhin exists then
        set pointer #1 e1 for vessel where vslhin = .vvslhin
        newpage
        if e1 eq 0 then
          *(vessel exists in database)
          set var vvslno to vslno in #1
          set var vvslname to vslname in #1
          set var vvslmake to vslmake in #1
          set var vvslmod to vslmodel in #1
          set var vvslyear to vslyear in #1
          set var vvslton to vsltons in #1
          set var vvslft to vsllenft in #1
          set var vvslin to vsllenin in #1
          set var vvslhp to vslhp in #1
          set var vvsluse to vsluse in #1
          set var vusedscr to vusedesc in boatuse where vuse eq +
          .vvsluse
          set var vvsltype to vsltype in #1
          set var vtypdscr to vtypedes in boattype where vtype eq +
          .vvsltype
          set var vvslprop to vslprop in #1
          set var vpropdsc to propdesc in boatprop where proptype eq +
          .vvslprop
          set var vvslhull to vslhull in #1
          set var vhulldsc to hmatdesc in hullmat where hmattype eq +
          .vvslhull
          set var vvsleng to vsleng in #1
          set var vengdesc to ecompdes in engcomp where ecomptyp eq +
          .vvsleng
          set var vvslfuel to vslfuel in #1
          set var vfueldes to fcompdes in fuelcomp where fcomptyp eq +
          .vvslfuel
          set var vconstr to vslconst in #1
          set var vcondesc to consdesc in constr where constype eq +
          .vconstr
          set var vdecal to cmedecal in #1
          set var vcmeyear to cmeyear in #1
          set var voffno to vsloffno in #1
          set var vhome to homeport in #1
          set var vcall to vslcall in #1
          set var vvalue to vslvalue in #1
          set var vsuper to vslsuper in #1
          set var vmasts to vslmasts in #1
          set var vsailno to sailno in #1
          set var vhlcolor to hlcolor in #1
          set var vtrcolor to trcolor in #1
          set var vslcolor to slcolor in #1
          set var vcbcolor to cbcolor in #1
          set var vdkcolor to dkcolor in #1
          set var vmarks to vslmarks in #1
          newpage
          draw vslquery with all at 1
          set var vslprn to A
          while vslprn ne P and vslprn ne C and vslprn ne M then
            write "Enter P to print vessel data, C to continue or M +
for main menu: " at 23 5
            fillin vslprn using " " at 23 68
            if vslprn eq P then
              output printer
              newpage
              print vslqprn
              output screen
            endif
            if vslprn eq M then
              set var menu to Y
              set var chk-hin to N
              break
            endif
          endwhile
          if menu eq Y then
            set var choice to 0
            break
          endif
          set pointer #2 e2 for vslintel where intvname eq .vvslname
          set var count to 1
          if e2 eq 0 then
            newpage
            write "US Coast Guard Group Monterey Intelligence History" +
            at 1 10
            write "-----------------------------------------------------+
---------------------------" at 2 1
          else
            newpage
            write "There is no intelligence for this vessel name" +
            at 10 10
            write "Press any key to continue" at 23 10
            pause
            newpage
          endif
          while e2 eq 0 then
            set var vintdate to intdate in #2
            set var vepic to epiccode in #2
            set var vstolen to stolen in #2
            set var vvrem1 to vremark1 in #2
            set var vvrem2 to vremark2 in #2
            set var vvrem3 to vremark3 in #2
            if count eq 1 then
              draw vintquer with all at 3
```

```
  endif
  if count eq 2 then
    draw vintquer with all at 9
  endif
  if count eq 3 then
    draw vintquer with all at 15
    write "Press any key to continue" at 23 10
    pause
    newpage
  endif
  set var count to .count + 1
  if count eq 4 then
    set var count to 1
  endif
  next #2 e2
endwhile
set pointer #2 e2 for vslintel where intvname eq .vvslname
if e2 eq 0 then
  write "Print vessel intelligence information? (Y)" at 23 10
  fillin vintprn using "(" at 23 49
  if vintprn fails or vintprn eq Y then
    newpage
    write "Preparing report, please stand by ........" +
    at 10 10
    remove vslint
    project vslint from vslintel using all sorted by intdate +
    where intvname eq .vvslname
    output printer
    newpage
    print vintqprn
    output screen
    remove vslint
  endif
endif
set pointer #2 e2 for boarding where bovhin eq .vvslhin
while e2 eq 0 then
  set var vbodate to bodate in #2
  set var vbotime to botime in #2
  set var vbono to bono in #2
  set var vvslpob to vslpob in #2
  set var vbolat to obslat in #2
  set var vbolong to obslong in #2
  set var vviol54 to viol54 in #2
  set var vviol55 to viol55 in #2
  set var vviol56 to viol56 in #2
  set var vviol57 to viol57 in #2
  set var vviol58 to viol58 in #2
  set var vviol59 to viol59 in #2
  set var vviol60 to viol60 in #2
  set var vviol61 to viol61 in #2
  set var vviol62 to viol62 in #2
  set var vviol63 to viol63 in #2
  set var vviol64 to viol64 in #2
  set var vviol65 to viol65 in #2
  set var vviol66 to viol66 in #2
  set var vviol67 to viol67 in #2
  set var vviol68 to viol68 in #2
  set var vuns69 to uns69 in #2
  set var vuns70 to uns70 in #2
  set var vuns71 to uns71 in #2
  set var vuns72 to uns72 in #2
  set var vuns73 to uns73 in #2
  set var vuns74 to uns74 in #2
  set var vuns75 to uns75 in #2
  set var vuns76 to uns76 in #2
  set var vuns77 to uns77 in #2
  set var vuns78 to uns78 in #2
  set var vuns79 to uns79 in #2
  set var vborem1 to borem1 in #2
  set var vborem2 to borem2 in #2
```

```
            set var vborem3 to borem3 in #2
            set var vboname to boname in #2
            set var vborate to borate in #2
            set var vweapons to weapons in #2
            set var valc to uns76 in #2
            newpage
            write "US Coast Guard Group Monterey Boarding History" +
            at 1 10
            write "------------------------------------------------+
------------------------------" at 2 1
            draw brdquery with all at 3
            write "Press any key to continue" at 23 10
            pause
            next #2 e2
          endwhile
          set pointer #2 e2 for boarding where bovhin eq .vvslhin
          if e2 eq 0 then
            write "Print boarding history? (Y)" at 23 10
            fillin bohisprn using "(" at 23 34
            if bohisprn fails or bohisprn eq Y then
              newpage
              write "Preparing report, please stand by ........" +
              at 10 10
              remove bohis
              project bohis from boarding using bodate botime bono +
              vslpob obslat obslong viol54 viol55 viol56 viol57 viol58 +
              viol59 viol60 viol61 viol62 viol63 viol64 viol65 viol66 +
              viol67 viol68 uns69 uns70 uns71 uns72 uns73 uns74 uns75 +
              uns76 uns77 uns78 uns79 borem1 borem2 borem3 boname +
              borate weapons bovhin sorted by bodate where +
              bovhin eq .vvslhin
              output printer
              newpage
              print boqprn
              output screen
              remove bohis
            endif
          endif
          if e2 ne 0 then
            newpage
            write "There are no boarding cases for this vessel" +
            at 10 10
            write "Press any key to continue" at 20 10
            pause
          endif
          set pointer #2 e2 for sar where sarvhin eq .vvslhin
          set var count to 1
          if e2 eq 0 then
            newpage
            write "US Coast Guard Group Monterey Sar History" at 1 10
            write "--------------------------------------------------+
-----------------------------" at 2 1
          endif
          while e2 eq 0 then
            set var vfoldno to folderno in #2
            set var vucn to ucn in #2
            set var vmucn to mucn in #2
            set var vsardate to sardate in #2
            set var vsarday to sarday in #2
            set var vsartime to sartime in #2
            set var vsarnod to sarnod in #2
            set var vsarpob to sarpob in #2
            set var vsarlat to sarlat in #2
            set var vsarlong to sarlong in #2
            set var vdistoff to offshore in #2
            set var vimmed to immedsar in #2
            if count eq 1 then
              draw sarquery with all at 3
            endif
            if count eq 2 then
```

```
              draw sarquery with all at 9
            endif
            if count eq 3 then
              draw sarquery with all at 15
              write "Press any key to continue" at 23 10
              pause
              newpage
            endif
            set var count to .count + 1
            if count eq 4 then
              set var count to 1
            endif
            next #2 e2
          endwhile
          set pointer #2 e2 for sar where sarvhin eq .vvslhin
          if e2 eq 0 then
            write "Print sar history? (Y)" at 23 10
            fillin sarhsprn using "(" at 23 29
            if sarhsprn fails or sarhsprn eq Y then
              newpage
              write "Preparing report, please stand by ........" +
              at 10 10
              remove sarhis
              project sarhis from sar using folderno ucn mucn sardate +
              sartime sarnod sarpob sarlat sarlong offshore immedsar +
              sarvhin sorted by sardate where sarvhin eq .vvslhin
              output printer
              newpage
              print sarqprn
              output screen
              remove sarhis
            endif
          endif
          if e2 ne 0 then
            newpage
            write "There are no sar cases for this vessel" at 10 10
            write "Press any key to continue" at 20 10
            pause
          endif
          set pointer #2 e2 for owvsl where vhinown eq .vvslhin
          newpage
          while e2 eq 0 then
            set var vowid to idowner in #2
            set pointer #3 e3 for owners where ownid eq .vowid
            set var vowlname to ownlname in #3
            set var vowfname to ownfname in #3
            set var vowmi to ownmi in #3
            set var vowaddr to ownaddr in #3
            set var vowcity to owncity in #3
            set var vowst to ownstate in #3
            set var vowzip to ownzip in #3
            set var vowtelno to owntelno in #3
            set var vowdob to owndob in #3
            newpage
            draw ownquery with all
            write "Press any key to continue" at 23 10
            pause
            set pointer #1 e1 for perintel where intlname eq .vowlname
            set var count to 1
            if e1 eq 0 then
              newpage
              write "US Coast Guard Group Monterey Intelligence +
History" at 1 10
              write "--------------------------------------------+
--------------------------------------" at 2 1
            else
              newpage
              write "There is no intelligence for this owner" at 10 10
            endif
            while e1 eq 0 then
```

```
    set var vlname to intlname in #1
    set var vfname to intfname in #1
    set var vmi to intmi in #1
    set var vdob to intdob in #1
    set var vrem1 to pintrem1 in #1
    set var vrem2 to pintrem2 in #1
    set var vrem3 to pintrem3 in #1
    set var vdate to pintdate in #1
    if count eq 1 then
      draw pintquer with all at 3
    endif
    if count eq 2 then
      draw pintquer with all at 9
    endif
    if count eq 3 then
      draw pintquer with all at 15
      write "Press any key to continue                 " +
      at 23 10
      pause
      newpage
    endif
    set var count to .count + 1
    if count eq 4 then
      set var count to 1
    endif
    next #1 e1
  endwhile
  write "Press any key to continue                      " +
  at 23 10
  pause
  next #2 e2
endwhile
write "Print owners and owner intelligence? (Y)" at 23 10
fillin owhisprn using "(" at 23 47
if owhisprn fails or owhisprn eq Y then
  newpage
  write "Preparing report, please stand by ........" at 10 10
  set pointer #2 e2 for owvsl where vhinown eq .vvslhin
  while e2 eq 0 then
    set var vowid to idowner in #2
    set pointer #3 e3 for owners where ownid eq .vowid
    set var vowlname to ownlname in #3
    set var vowfname to ownfname in #3
    set var vowmi to ownmi in #3
    set var vowaddr to ownaddr in #3
    set var vowcity to owncity in #3
    set var vowst to ownstate in #3
    set var vowzip to ownzip in #3
    set var vowtelno to owntelno in #3
    set var vowdob to owndob in #3
    output printer
    newpage
    print owqprn
    output screen
    remove perint
    project perint from perintel using all sorted by +
    pintdate where intlname eq .vowlname
    output printer
    newpage
    print pintqprn
    output screen
    next #2 e2
  endwhile
  remove perint
endif
set pointer #2 e2 for opvsl where vhinoper eq .vvslhin
newpage
while e2 eq 0 then
  set var vopid to idoper in #2
  set pointer #3 e3 for operator where opid eq .vopid
  set var voplname to oplname in #3
  set var vopfname to opfname in #3
  set var vopmi to opmi in #3
  set var vopaddr to opaddr in #3
  set var vopcity to opcity in #3
  set var vopst to opstate in #3
  set var vopzip to opzip in #3
  set var voptelno to optelno in #3
  set var vopdob to opdob in #3
  set var vopcrse to opcourse in #3
  newpage
  draw opquery with all
  write "Press any key to continue" at 23 10
  pause
  set pointer #1 e1 for perintel where intlname eq .voplname
  set var count to 1
  if e1 eq 0 then
    newpage
    write "US Coast Guard Group Monterey Intelligence +
    History" at 1 10
    write "-----------------------------------------------+
-----------------------------------" at 2 1
  else
    newpage
    write "There is no intelligence for this operator" +
    at 10 10
  endif
  while e1 eq 0 then
    set var vlname to intlname in #1
    set var vfname to intfname in #1
    set var vmi to intmi in #1
    set var vdob to intdob in #1
    set var vrem1 to pintrem1 in #1
    set var vrem2 to pintrem2 in #1
    set var vrem3 to pintrem3 in #1
    set var vdate to pintdate in #1
    if count eq 1 then
      draw pintquer with all at 3
    endif
    if count eq 2 then
      draw pintquer with all at 9
    endif
    if count eq 3 then
      draw pintquer with all at 15
      write "Press any key to continue" at 23 10
      pause
      newpage
    endif
    set var count to .count + 1
    if count eq 4 then
      set var count to 1
    endif
    next #1 e1
  endwhile
  write "Press any key to continue                  " at 23 10
  pause
  next #2 e2
endwhile
write "Print operators and operator intelligence? (Y)" +
at 23 10
fillin ophisprn using "(" at 23 53
if ophisprn fails or ophisprn eq Y then
  newpage
  write "Preparing report, please stand by ........" at 10 10
  set pointer #2 e2 for opvsl where vhinoper eq .vvslhin
  while e2 eq 0 then
    set var vopid to idoper in #2
    set pointer #3 e3 for operator where opid eq .vopid
    set var voplname to oplname in #3
    set var vopfname to opfname in #3
```

```
              set var vopmi to opmi in #3
              set var vopaddr to opaddr in #3
              set var vopcity to opcity in #3
              set var vopst to opstate in #3
              set var vopzip to opzip in #3
              set var voptelno to optelno in #3
              set var vopdob to opdob in #3
              set var vopcrse to opcourse in #3
              output printer
              newpage
              print opqprn
              output screen
              remove perint
              project perint from perintel using all sorted by +
              pintdate where intlname eq .voplname
              output printer
              newpage
              print pintqprn
              output screen
              next #2 e2
            endwhile
            remove perint
          endif
          newpage
          write "End of data for this vessel" at 21 10
          set var chk-hin to N
          break
        else
          newpage
          write "This HIN does not exist in the database" at 10 20
          write "Do you wish to reenter the HIN? (Y)" at 12 20
          fillin correct using "(" at 12 52
          if correct exists and correct eq N then
            set var chk_hin to N
          endif
          break
        endif
      endwhile
    endwhile
  endif
  if choice eq 5 then
    set var chk-intl to Y
    while chk-intl eq Y then
      newpage
      write "Enter name of desired vessel: " at 10 10
      fillin vvslname using " " at 10 39
      while vvslname exists then
        set pointer #2 e2 for vslintel where intvname eq .vvslname
        set var count to 1
        if e2 eq 0 then
          newpage
          write "US Coast Guard Group Monterey Intelligence History" +
          at 1 10
          write "---------------------------------------------------+
---------------------------" at 2 1
        else
          newpage
          write "There is no intelligence for this vessel name" at 10 10
          write "Press any key to continue" at 23 10
          pause
        endif
        while e2 eq 0 then
          set var vintdate to intdate in #2
          set var vepic to epiccode in #2
          set var vstolen to stolen in #2
          set var vvrem1 to vremark1 in #2
          set var vvrem2 to vremark2 in #2
          set var vvrem3 to vremark3 in #2
          if count eq 1 then
            draw vintquer with all at 3
          endif
          if count eq 2 then
            draw vintquer with all at 9
          endif
          if count eq 3 then
            draw vintquer with all at 15
            write "Press any key to continue" at 23 10
            pause
            newpage
          endif
          set var count to .count + 1
          if count eq 4 then
            set var count to 1
          endif
          next #2 e2
        endwhile
        set pointer #2 e2 for vslintel where intvname eq .vvslname
        if e2 eq 0 then
          write "Print vessel intelligence information? (Y)" at 23 10
          fillin vintprn using "(" at 23 49
          if vintprn fails or vintprn eq Y then
            remove vslint
            project vslint from vslintel using all sorted by intdate +
            where intvname eq .vvslname
            output printer
            newpage
            print vintqprn
            output screen
            remove vslint
          endif
        endif
        write "End of data for this vessel"
        set var chk-intl to N
      endwhile
    endwhile
  endif
  if choice eq 6 then
    set var more to N
  endif
endwhile  *(end of main loop)
remove temp
remove bohis
remove sarhis
remove perint
remove vslint
set null -0-
return
*( end )


$COMMAND
QUER_PER
*(**********************************************************************

    Program:              Query Personnel
    Author:               H R Lang and J D Allen
    Date Written:         August 1987
    Description:          This program searches database for all
                          information concerning a particular person,
                          including all vessels owned or operated
                          and all historical information concerning
                          these vessels.
    Tables Used:          vessel, sar, boarding, owner, operator,
                          owvsl, opvsl, opown, vslintel, perintel
    Temp Tables:          temp, vslint, perint, bohis, sarhis,
                          vslsown, vslsopr, wvsljoin, pvsljoin
    Reports Used:         vslqprn, boqprn, sarqprn, perqprn,
                          pintqprn, vintqprn
    Forms Used:           vslquery, brdquery, sarquery, perquer1,
                      perquer2, pintquer, vintquer

**********************************************************************)

set messages off
set escape off
newpage
write "Program loading in progress.  Please stand by .........." +
at 10 10
clear all variables
set null " "
set var more to Y
remove temp  *(relation for print purposes only)
define
attributes
tmp text 2
relations
temp with tmp
end
load temp
  "XX"
end
while more eq Y then  *(main loop)
  set var ownfound to N
  set var oprfound to N
  set var status to N
  newpage
  write "Press [PGDN] after entering initial information to check if" +
  at 1 5
  write "person exists in database.  Press [ESC] to quit" at 2 10
  draw perquer1 at 4
  enter var return esc pgdn
  if #return ne esc then
    if vlname exists and vfname exists then
      set pointer #2 e2 for owners where ownlname contains .vlname and +
      ownfname contains .vfname
      newpage
      while e2 eq 0 then
        *(last/first name exist in owners)
        set var vplname to ownlname in #2
        set var vpfname to ownfname in #2
        set var vpermi to ownmi in #2
        set var vperdob to owndob in #2
        set var vperaddr to ownaddr in #2
        set var vpercity to owncity in #2
        set var vperst to ownstate in #2
        set var vperzip to ownzip in #2
        set var vpertel to owntelno in #2
        newpage
        draw perquer2 with all at 3
        write "Correct owner? (N)" at 21 20
        fillin owfound using "(" at 21 35
        if owfound exists and owfound eq Y then
          set var vowid integer
          set var vowid to ownid in #2
          set var ownfound to Y
          break
        endif
        next #2 e2
      endwhile
      set var chk-op to Y
      if ownfound eq Y then
        set pointer #2 e2 for opown where ownno eq .vowid
        if e2 eq 0 then
          set var vopid to opno in #2
          set var oprfound to Y
          set var chk-op to N
        else
          set pointer #3 e3 for operator where oplname = .vlname and +
    opfname = .vfname
    *(check if an operator record = owner record)
    newpage
    while e3 eq 0 then
      *(last/first name exist in operator)
      set var voplname to oplname in #3
      set var vopfname to opfname in #3
      set var vopmi to opmi in #3
      set var vopdob to opdob in #3
      set var vopaddr to opaddr in #3
      set var vopcity to opcity in #3
      set var vopst to opstate in #3
      set var vopzip to opzip in #3
      set var voptelno to optelno in #3
      set var owoptest to T
      if vplname ne .voplname or vpfname ne .vopfname then
        set var owoptest to F
      endif
      if vpermi exists and vopmi exists and vpermi ne .vopmi then
        set var owoptest to F
      endif
      if vperdob exists and vopdob exists and vperdob ne .vopdob +
      then
        set var owoptest to F
      endif
      if vperaddr exists and vopaddr exists and vperaddr ne +
      .vopaddr then
        set var owoptest to F
      endif
      if vpercity exists and vopcity exists and vpercity ne +
      .vopcity then
        set var owoptest to F
      endif
      if vperst exists and vopst exists and vperst ne .vopst then
        set var owoptest to F
      endif
      if vperzip exists and vopzip exists and vperzip ne .vopzip +
      then
        set var owoptest to F
      endif
      if vpertel exists and voptelno exists and vpertel ne +
      .voptelno then
        set var owoptest to F
      endif
      if owoptest eq T then
        set var vopid integer
        set var vopid to opid in #3
        set var chk-op to N
        set var oprfound to Y
        break
      endif
      next #3 e3
    endwhile
  endif
endif
if chk-op eq Y then
  set pointer #3 e3 for operator where oplname contains .vlname +
  and opfname contains .vfname
  newpage
  while e3 eq 0 then
    *(last/first name exist in operator)
    set var vplname to oplname in #3
    set var vpfname to opfname in #3
    set var vpermi to opmi in #3
    set var vperdob to opdob in #3
    set var vperaddr to opaddr in #3
    set var vpercity to opcity in #3
    set var vperst to opstate in #3
    set var vperzip to opzip in #3
    set var vpertel to optelno in #3
```

```
          newpage
          draw perquer2 with all at 3
          write "Correct operator? (N)" at 22 20
          fillin opfound using "(" at 22 38
          if opfound exists and opfound eq Y then
            set var vopid integer
            set var vopid to opid in #2
            set var oprfound to Y
            break
          endif
          next #3 e3
        endwhile
      endif
      if ownfound eq Y or oprfound eq Y then
        newpage
        write "Print owner/operator information? (Y)" at 10 10
        fillin persprn using "(" at 10 44
        if persprn fails or persprn eq Y then
          newpage
          write "Preparing report, please stand by........." at 10 10
          output printer
          newpage
          print perqprn
          output terminal
        endif
      endif
      if ownfound eq N and oprfound eq N then
        set var vplname to .vlname
      endif
      set pointer #1 e1 for perintel where intlname contains .vplname
      set var count to 1
      if e1 eq 0 then
        newpage
        write "US Coast Guard Group Monterey Intelligence History" +
        at 1 10
        write "---------------------------------------------------+
--------------------------" at 2 1
      else
        newpage
        write "There is no intelligence for this person" at 10 10
        write "Press any key to continue                   " at 23 10
        pause
      endif
      while e1 eq 0 then
        set var vlname to intlname in #1
        set var vfname to intfname in #1
        set var vmi to intmi in #1
        set var vdob to intdob in #1
        set var vrem1 to pintrem1 in #1
        set var vrem2 to pintrem2 in #1
        set var vrem3 to pintrem3 in #1
        set var vdate to pintdate in #1
        if count eq 1 then
          draw pintquer with all at 3
        endif
        if count eq 2 then
          draw pintquer with all at 9
        endif
        if count eq 3 then
          draw pintquer with all at 15
          write "Press any key to continue                " at 23 10
          pause
          newpage
        endif
        set var count to .count + 1
        if count eq 4 then
          set var count to 1
        endif
        next #1 e1
      endwhile
```

```
set pointer #1 el for perintel where intlname eq .vplname
if el eq 0 then
  write "Print personnel intelligence? (Y)" at 23 10
  fillin perintpr using "(" at 23 40
  if perintpr fails or perintpr eq Y then
    newpage
    write "Preparing report, please stand by........." at 10 10
    remove perint
    project perint from perintel using all sorted by pintdate +
    where intlname eq .vplname
    output printer
    newpage
    print pintqprn
    output screen
    remove perint
  endif
endif
*(determine status of person, select vessels owned and operated)
*(W = owner, P = operator, B = both and N = neither)
if ownfound eq Y and oprfound eq Y then
  set var status to B
  remove vslsown
  project vslsown from owvsl using all where idowner eq .vowid
  remove wvsljoin
  join vslsown using vhinown with vessel using vslhin forming +
  wvsljoin where eq
  newpage
  write "This individual owns the following vessels:" at 1 2
  write " "
  sel vslhin vslname from wvsljoin
  write " "
  write "End of data."
  remove vslsopr
  project vslsopr from opvsl using all where idoper eq .vopid
  remove pvsljoin
  join vslsown using vhinown with vessel using vslhin forming +
  pvsljoin where eq
  write " "
  write " "
  write "This individual has operated the following vessels:"
  write " "
  sel vslhin vslname from pvsljoin
  write " "
  write "End of data. Press any key to continue"
  pause
endif
if ownfound eq Y and oprfound eq N then
  set var status to W
  remove vslsown
  project vslsown from owvsl using all where idowner eq .vowid
  remove wvsljoin
  join vslsown using vhinown with vessel using vslhin forming +
  wvsljoin where eq
  newpage
  write "This individual owns the following vessels:" at 1 10
  write " "
  sel vslhin vslname from wvsljoin
  write " "
  write "End of data. Press any key to continue"
  pause
endif
if ownfound eq N and oprfound eq Y then
  set var status to P
  remove vslsopr
  project vslsopr from opvsl using all where idoper eq .vopid
  remove pvsljoin
  join vslsown using vhinown with vessel using vslhin forming +
  pvsljoin where eq
  newpage
  write "This individual has operated the following vessels:" +
        at 1 10
        write " "
        sel vslhin vslname from pvsljoin
        write " "
        write "End of data. Press any key to continue"
        pause
      endif
      if status ne N then
       newpage
       write "Do you wish to search for full information concerning +
these vessels? (N)" at 10 5
       fillin searchok using "(" at 10 74
       if searchok exists and searchok eq Y then
        set var done to N
        set var owndone to N
        while done eq N then
          if status eq B and owndone eq N then
            remove vslsown
            project vslsown from owvsl using all where idowner eq .vowid
            set pointer #3 e3 for vslsown
          endif
          if status eq W then
            remove vslsown
            project vslsown from owvsl using all where idowner eq .vowid
            set pointer #3 e3 for vslsown
          endif
          if status eq P then
            remove vslsopr
            project vslsopr from opvsl using all where idoper eq .vopid
            set pointer #3 e3 for vslsopr
          endif
          if status eq B and owndone eq Y then
            remove vslsopr
            project vslsopr from opvsl using all where idoper eq .vopid
            set pointer #3 e3 for vslsopr where vhinoper nea vhinown +
            in vslsown
          endif
          while e3 eq 0 then
            if status eq B and owndone eq N or status eq W then
              set var vvslhin to vhinown in #3
            endif
            if status eq B and owndone eq Y or status eq P then
              set var vvslhin to vhinoper in #3
            endif
            set pointer #1 e1 for vessel where vslhin eq .vvslhin
            set var vvslno to vslno in #1
            set var vvslname to vslname in #1
            set var vvslmake to vslmake in #1
            set var vvslmod to vslmodel in #1
            set var vvslyear to vslyear in #1
            set var vvslton to vsltons in #1
            set var vvslft to vsllenft in #1
            set var vvslin to vsllenin in #1
            set var vvslhp to vslhp in #1
            set var vvsluse to vsluse in #1
            set var vusedscr to vusedesc in boatuse where vuse eq +
            .vvsluse
            set var vvsltype to vsltype in #1
            set var vtypdscr to vtypedes in boattype where vtype eq +
            .vvsltype
            set var vvslprop to vslprop in #1
            set var vpropdsc to propdesc in boatprop where proptype eq +
            .vvslprop
            set var vvslhull to vslhull in #1
            set var vhulldsc to hmatdesc in hullmat where hmattype eq +
            .vvslhull
            set var vvsleng to vsleng in #1
            set var vengdesc to ecompdes in engcomp where ecomptyp eq +
            .vvsleng
            set var vvslfuel to vslfuel in #1
            set var vfueldes to fcompdes in fuelcomp where fcomptyp eq +
            .vvslfuel
            set var vconstr to vslconst in #1
            set var vcondesc to consdesc in constr where constype eq +
            .vconstr
            set var vdecal to cmedecal in #1
            set var vcmeyear to cmeyear in #1
            set var voffno to vsloffno in #1
            set var vhome to homeport in #1
            set var vcall to vslcall in #1
            set var vvalue to vslvalue in #1
            set var vsuper to vslsuper in #1
            set var vmasts to vslmasts in #1
            set var vsailno to sailno in #1
            set var vhlcolor to hlcolor in #1
            set var vtrcolor to trcolor in #1
            set var vslcolor to slcolor in #1
            set var vcbcolor to cbcolor in #1
            set var vdkcolor to dkcolor in #1
            set var vmarks to vslmarks in #1
            newpage
            if status eq B and owndone eq N or status eq W then
              write "This vessel is owned by the designated +
individual.  Press any key to continue" at 1 1
            endif
            if status eq B and owndone eq Y or status eq P then
              write "Subject person has operated this vessel.  Press +
any key to continue" at 1 1
            endif
            draw vslquery with all at 3
            pause
            set pointer #2 e2 for vslintel where intvname eq .vvslname
            set var count to 1
            if e2 eq 0 then
              newpage
              write "US Coast Guard Group Monterey Intelligence +
History" at 1 10
              write "-----------------------------------------------+
---------------------------------" at 2 1
            else
              newpage
              write "There is no intelligence for this vessel name" +
              at 10 10
              write "Press any key to continue" at 23 10
              pause
            endif
            newpage
            while e2 eq 0 then
              set var vintdate to intdate in #2
              set var vepic to epiccode in #2
              set var vstolen to stolen in #2
              set var vvrem1 to vremark1 in #2
              set var vvrem2 to vremark2 in #2
              set var vvrem3 to vremark3 in #2
              if count eq 1 then
                draw vintquer with all at 3
              endif
              if count eq 2 then
                draw vintquer with all at 9
              endif
              if count eq 3 then
                draw vintquer with all at 15
                write "Press any key to continue" at 23 10
                pause
                newpage
              endif
              set var count to .count + 1
              if count eq 4 then
                set var count to 1
              endif
```

```
  next #2 e2
endwhile
if status eq W or status eq B then
  set pointer #3 e3 for vslsown
endif
if status eq P then
  set pointer #3 e3 for vslsopr
endif
if status eq B and owndone eq Y then
  set pointer #3 e3 for vslsopr where vhinoper nea +
  vhinown in vslsown
endif
if e3 eq 0 then
  write "Print vessel information and intelligence? (Y)" +
  at 23 10
  fillin vslprn using "(" at 23 53
  if vslprn fails or vslprn eq Y then
    newpage
    write "Preparing report, please stand by........." +
    at 10 10
    while e3 eq 0 then
      set var vvslhin to vhinown in #3
      set pointer #1 e1 for vessel where vslhin eq .vvslhin
      set var vvslno to vslno in #1
      set var vvslname to vslname in #1
      set var vvslmake to vslmake in #1
      set var vvslmod to vslmodel in #1
      set var vvslyear to vslyear in #1
      set var vvslton to vsltons in #1
      set var vvslft to vsllenft in #1
      set var vvslin to vsllenin in #1
      set var vvslhp to vslhp in #1
      set var vvsluse to vsluse in #1
      set var vusedscr to vusedesc in boatuse where vuse +
      eq .vvsluse
      set var vvsltype to vsltype in #1
      set var vtypdscr to vtypedes in boattype where +
      vtype eq .vvsltype
      set var vvslprop to vslprop in #1
      set var vpropdsc to propdesc in boatprop where +
      proptype eq .vvslprop
      set var vvslhull to vslhull in #1
      set var vhulldsc to hmatdesc in hullmat where +
      hmattype eq .vvslhull
      set var vvsleng to vsleng in #1
      set var vengdesc to ecompdes in engcomp where +
      ecomptyp eq .vvsleng
      set var vvslfuel to vslfuel in #1
      set var vfueldes to fcompdes in fuelcomp where +
      fcomptyp eq .vvslfuel
      set var vconstr to vslconst in #1
      set var vcondesc to consdesc in constr where +
      constype eq .vconstr
      set var vdecal to cmedecal in #1
      set var vcmeyear to cmeyear in #1
      set var voffno to vsloffno in #1
      set var vhome to homeport in #1
      set var vcall to vslcall in #1
      set var vvalue to vslvalue in #1
      set var vsuper to vslsuper in #1
      set var vmasts to vslmasts in #1
      set var vsailno to sailno in #1
      set var vhlcolor to hlcolor in #1
      set var vtrcolor to trcolor in #1
      set var vslcolor to slcolor in #1
      set var vcbcolor to cbcolor in #1
      set var vdkcolor to dkcolor in #1
      set var vmarks to vslmarks in #1
      newpage
      write "Preparing report, please stand by........." +
                 at 10 10
                 output printer
                 newpage
                 print vslqprn
                 output screen
                 set pointer #2 e2 for vslintel where intvname eq +
                 .vvslname
                 if e2 eq 0 then
                   remove vslint
                   project vslint from vslintel using all sorted by +
                   intdate where intvname eq .vvslname
                   output printer
                   newpage
                   print vintqprn
                   output screen
                   remove vslint
                 endif
                 next #3 e3
               endwhile
             endif
           endif
           set pointer #2 e2 for boarding where bovhin eq .vvslhin
           while e2 eq 0 then
             set var vbodate to bodate in #2
             set var vbotime to botime in #2
             set var vbono to bono in #2
             set var vvslpob to vslpob in #2
             set var vbolat to obslat in #2
             set var vbolong to obslong in #2
             set var vviol54 to viol54 in #2
             set var vviol55 to viol55 in #2
             set var vviol56 to viol56 in #2
             set var vviol57 to viol57 in #2
             set var vviol58 to viol58 in #2
             set var vviol59 to viol59 in #2
             set var vviol60 to viol60 in #2
             set var vviol61 to viol61 in #2
             set var vviol62 to viol62 in #2
             set var vviol63 to viol63 in #2
             set var vviol64 to viol64 in #2
             set var vviol65 to viol65 in #2
             set var vviol66 to viol66 in #2
             set var vviol67 to viol67 in #2
             set var vviol68 to viol68 in #2
             set var vuns69 to uns69 in #2
             set var vuns70 to uns70 in #2
             set var vuns71 to uns71 in #2
             set var vuns72 to uns72 in #2
             set var vuns73 to uns73 in #2
             set var vuns74 to uns74 in #2
             set var vuns75 to uns75 in #2
             set var vuns76 to uns76 in #2
             set var vuns77 to uns77 in #2
             set var vuns78 to uns78 in #2
             set var vuns79 to uns79 in #2
             set var vborem1 to borem1 in #2
             set var vborem2 to borem2 in #2
             set var vborem3 to borem3 in #2
             set var vboname to boname in #2
             set var vborate to borate in #2
             set var vweapons to weapons in #2
             set var valc to uns76 in #2
             newpage
             write "US Coast Guard Group Monterey Boarding History" +
             at 1 10
             write "----------------------------------------------+
-----------------------------" at 2 1
             draw brdquery with all at 3
             write "Press any key to continue" at 23 10
             pause
```

```
              next #2 e2
            endwhile
            set pointer #2 e2 for boarding where bovhin eq .vvslhin
            if e2 eq 0 then
              write "Print boarding history? (Y)" at 23 10
              fillin bohisprn using "(" at 23 34
              if bohisprn fails or bohisprn eq Y then
                newpage
                write "Preparing report, please stand by........" +
                at 10 10
                remove bohis
                project bohis from boarding using bodate botime bono +
                vslpob obslat obslong viol54 viol55 viol56 viol57 +
                viol58 viol59 viol60 viol61 viol62 viol63 viol64 +
                viol65 viol66 viol67 viol68 uns69 uns70 uns71 uns72 +
                uns73 uns74 uns75 uns76 uns77 uns78 uns79 borem1 +
                borem2 borem3 boname borate weapons bovhin sorted by +
                bodate where bovhin eq .vvslhin
                output printer
                newpage
                print boqprn
                output screen
                remove bohis
              endif
            endif
            if e2 ne 0 then
              newpage
              write "There are no boarding cases for this vessel" +
              at 10 10
              write "Press any key to continue" at 20 10
              pause
            endif
            set pointer #2 e2 for sar where sarvhin eq .vvslhin
            set var count to 1
            if e2 eq 0 then
              newpage
              write "US Coast Guard Group Monterey Sar History" +
              at 1 10
              write "--------------------------------------------+
----------------------------------" at 2 1
            endif
            while e2 eq 0 then
              set var vfoldno to folderno in #2
              set var vucn to ucn in #2
              set var vmucn to mucn in #2
              set var vsardate to sardate in #2
              set var vsarday to sarday in #2
              set var vsartime to sartime in #2
              set var vsarnod to sarnod in #2
              set var vsarpob to sarpob in #2
              set var vsarlat to sarlat in #2
              set var vsarlong to sarlong in #2
              set var vdistoff to offshore in #2
              set var vimmed to immedsar in #2
              if count eq 1 then
                draw sarquery with all at 3
              endif
              if count eq 2 then
                draw sarquery with all at 9
              endif
              if count eq 3 then
                draw sarquery with all at 15
                write "Press any key to continue" at 23 10
                pause
                newpage
              endif
              set var count to .count + 1
              if count eq 4 then
                set var count to 1
              endif
              next #2 e2
            endwhile
            set pointer #2 e2 for sar where sarvhin eq .vvslhin
            if e2 eq 0 then
              write "Print sar history? (Y)" at 23 10
              fillin sarhsprn using "(" at 23 29
              if sarhsprn fails or sarhsprn eq Y then
                newpage
                write "Preparing report, please stand by........." +
                at 10 10
                remove sarhis
                project sarhis from sar using folderno ucn mucn +
                sardate sartime sarnod sarpob sarlat sarlong offshore +
                immedsar sarvhin sorted by sardate where sarvhin eq +
                .vvslhin
                output printer
                newpage
                print sarqprn
                output screen
                remove sarhis
              endif
            endif
            if e2 ne 0 then
              newpage
              write "There are no sar cases for this vessel" at 10 10
              write "Press any key to continue" at 20 10
              pause
            endif
            next #3 e3
          endwhile
          if status eq W or status eq P then
            set var done to Y
          endif
          if status eq B and owndone eq N then
            set var owndone to Y
          endif
          if status eq B and owndone eq Y then
            set var done to Y
          endif
        endwhile
        remove vslsown
        remove vslsopr
        remove wvsljoin
        remove pvsljoin
       endif
      else
        newpage
        write "This name does not exist as an owner or an operator" +
        at 10 10
        write "Press any key to continue" at 15 15
        pause
      endif
    endif
  endif
  label moreblk
  newpage
  write "More names to check? (N)" at 10 10
  fillin mrname using "(" at 10 31
  if mrname fails or mrname eq N then
    set var more to N
  else
    clear all variables
    set var more to Y
  endif
endwhile  *(end of main loop)
remove temp
remove vslsown
remove vslsopr
remove perint
remove vslint
remove bohis
remove sarhis
set null -0-
return
*( end )
```

APPENDIX P
SUTILITY SOURCE CODE

```
$COMMAND
UTLMAIN
*(**********************************************************************

                    US COAST GUARD GROUP MONTEREY
       SEARCH & RESCUE and ENFORCEMENT of LAWS and TREATIES
                   INTELLIGENCE SYSTEM (SARELTIS)

              UTILITIES MENU and APPLICATION PROGRAMS

                 Author:    Jon D. Allen, LT, USCG
                            Heidi R. Lang, LT, USN
                            AUGUST 1987

**********************************************************************)

set error messages off
set messages off
set escape off
clear all variables
newpage
set var pick11 integer
choose pick11 from utl_menu in sutility.apx
if pick11 eq 0 then
   quit to restutly.cmd
endif
if pick11 eq 1 then
   run INITIAL in sutility.apx
   quit to restutly.cmd
endif
if pick11 eq 2 then
   run PACKDBMS in sutility.apx
   quit to restutly.cmd
endif
if pick11 eq 3 then
   run BACKDBMS in sutility.apx
   quit to restutly.cmd
endif
if pick11 eq 4 then
   run RESTDBMS in sutility.apx
   quit to restutly.cmd
endif
if pick11 eq 5 then
   quit to restmain.cmd
endif
quit to restutly.cmd
return
$menu
utl_menu
column SARELTIS --- UTILITIES MENU
INITIALIZE DATABASE
PACK DATABASE
BACKUP DATABASE
RESTORE DATABASE
RETURN TO MAIN MENU
*( end )
```

```
$COMMAND
INITIAL
*(**********************************************************************

    Program:           Initialize SARSYS
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program removes all temporary
                       tables that can be created by the system.
    Tables Used:       none
    Temp Tables:       all
    Reports Used:      none
    Forms Used:        none

**********************************************************************)

set error messages off
set messages off
set escape off
newpage
remove epicsum
remove stolesum
remove intelsum
remove epictmp1
remove epictmp2
remove epictmp
remove sarsum
remove temp
remove bosum
remove vslsum
remove violsum
remove vslmo
remove summary1
remove summary2
remove summary3
remove summary4
remove summary5
remove summary6
remove summary7
remove frqchart
remove positfrq
remove sarsum
remove bovslsum
remove ownsum1
remove ownsum2
remove opsum1
remove opsum2
remove vslsum
remove vslsum1
remove vslsum2
remove sartime1
remove sartime2
remove sartime3
remove sartime4
remove sartime5
remove sarsum1
remove sarstat
remove intelsum
remove opvsls1
remove opvsls2
remove opervsls
remove ownvsls1
remove ownvsls2
remove ownvsls
newpage
return
*( end )
```

```
$COMMAND
PACKDBMS
*(***********************************************************************

    Program:           Pack Database
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program packs/compresses the database.
    Tables Used:       none
    Temp Tables:       none
    Reports Used:      none
    Forms Used:        none

***********************************************************************)

set error messages off
set messages off
set escape off
clear all variables
newpage
write "Before packing the database,  please ensure that you have +
made" at 4 1
write "a back up of the database.  Ideally,  you should back up +
the " at 5 1
write "database just prior to performing the packing operation." at 6 1
set var choice text
write "Do you wish to continue with the packing program? (N)" at 8 1
set error messages on
fillin choice using "(" at 8 51
set error messages off
if choice fails or choice eq n then
   return
else
   write "Packing/Compressing the database will take several +
minutes." at 10 1
   write "Please be patient......." at 12 1
   write "Please press any key to continue" at 16 1
   pause
   newpage
   write "Now packing the database.........." at 10 1
   set error var erv
   pack sarsys
   if erv = 0 then
      write "Packing has successfully completed." at 12 1
   else
      write "An error has occurred during the packing." at 12 1
      write "Please refer to the reference manual page 5-25." at 13 1
   endif
   write "Please press any key to continue" at 18 1
endif
clear all variables
return
*( end )


$COMMAND
BACKDBMS
*(***********************************************************************

    Program:           Backup Database
    Author:            J D Allen and H R Lang
    Date Written:      August 1987
    Description:       This program provides instructions for
                       backing up the database.
    Tables Used:       none
    Temp Tables:       none
    Reports Used:      none
    Forms Used:        none

*************************************************************************)

set error messages off
set messages off
set escape off
newpage
write "Backing up of the database should be done at a minimum of once +
per" at 3 1
write "week.  To back the database up only requires that you exit +
the " at 4 1
write "database and executing a DOS (Disk Operating System) batch +
program" at 5 1
write "that has been specifically customized to do the backup for +
you." at 6 1
write "To exit the system,  you must choose the [EXIT SARSYS] option +
from" at 10 1
write "the master menu.  This will exit from RBase 5000 and return +
you to" at 11 1
write "the DOS prompt.  At the DOS prompt,  enter the following:" +
at 12 1
write "                  BACKDBMS                                  " +
at 14 1
write "Please follow the instructions/prompts that the computer +
displays" at 16 1
write "with regard to the insertion of formatted disks.  Upon +
completion" at 17 1
write "of the program,  you will automatically be returned to RBase +
5000." at 18 1
write "Please press any key when you are ready to proceed......+
at 23 1
pause
newpage
return
*( end )


$COMMAND
RESTDBMS
*(************************************************************************

    Program:          Restore Database
    Author:           J D Allen and H R Lang
    Date Written:     August 1987
    Description:      This program provides instructions for
                      restoring the database from a backup.
    Tables Used:      none
    Temp Tables:      none
    Reports Used:     none
    Forms Used:       none

*************************************************************************)

set error messages off
set messages off
set escape off
newpage
write "To restore the database only requires that you exit the " +
      at 4 1
write "database and execute a DOS (Disk Operating System) batch +
program" at 5 1
write "that has been specifically customized to do the restore for +
you." at 6 1
write "To exit the system,  you must choose the [EXIT SARSYS] option +
from" at 10 1
write "the master menu.  This will exit from RBase 5000 and return +
you to" at 11 1
write "the DOS prompt.  At the DOS prompt,  enter the following:" +
at 12 1
write "                    RESTDBMS                              " +
at 14 1
write "Please follow the instructions/prompts that the computer +
displays" at 16 1
write "with regard to the insertion of backup disks.  Upon completion" +
at 17 1
write "of the program,  you will automatically be returned to RBase +
5000." at 18 1
write "Please press any key when you are ready to proceed.........+
at 23 1
pause
newpage
return
*( end )
```

LIST OF REFERENCES

1. Commandant, U. S. Coast Guard Instruction 7100.1A, 11 February 1982.

2. Davis, William S., *Systems Analysis and Design*, Addison-Wesley Publishing Company, 1983.

3. Kroenke, David M., *DATABASE PROCESSING: Fundamentals, Design, Implementation*, Science Research Associates, Inc., 1983.

4. Graham, James M., *The Completed Management Information System for the Monterey Navy Flying Club*, Masters Thesis, Naval Postgraduate School, Monterey, California, March 1987.

5. *R:base 5000 User's Manual*, MicroRim, Inc., 1985.

BIBLIOGRAPHY

Ackoff, Russel L., "Management Misinformation Systems", *Management Science*, v. 14, no. 4, December 1967, pp. (B)147-(B)156.

Bloomfield, Howard V. L., *The Compact History of the United States Coast Guard*, Hawthorne Books, Inc., 1966.

Bohl, Marilyn, *Information Processing*, Science Research Associates, Inc., 1984.

Briggs, Charles L., Birkes, Evan G. and Atkins, William, *Managing the Systems Development Process*, Touche Ross and Company, 1980.

Coast Guard Fact File (1986-1987), Community Relations Branch (G-BPA-3), U. S. Coast Guard, July 1986.

Coast Guard Group Monterey Instruction 5400, 15 November 1984.

Commandant, U. S. Coast Guard Instruction 7100.1A, 11 February 1982.

Commandant, U. S. Coast Guard Instruction 16203.1, 29 May 1980.

Commandant, U. S. Coast Guard Instruction M16750.4, 13 September 1979.

Davis, William S., *Systems Analysis and Design*, Addison-Wesley Publishing Company, Inc., 1983.

Department of Transportation, U. S. Coast Guard, *Enforcement of Laws and Treaties FY 82-91*, Operating Program Plan (1979).

Department of Transportation, U. S. Coast Guard, *Missions of the United States Coast Guard (CG-378-8)*, 1985.

Dickson, Gary W. and Wetherbe, James C., *The Management of Information Systems*, McGraw-Hill Book Company, 1985.

Dinerstein, Nelson T., *R:base 5000 for the Programmer*, Scott, Foresman and Company, 1986.

Evans, Stephan H., *The United States Coast Guard 1790-1915*, The United States Naval Institute, 1949.

Graham, James M., *The Completed Management Information System for the Monterey Navy Flying Club*, Masters Thesis, Naval Postgraduate School, Monterey, California, March 1987.

Kent, William, "A Simple Guide to Five Normal Forms in Relational Database Theory", *Communications of the ACM*, vol 26, no. 2, February 1983, pp. 120-125.

King, David, *Current Practices in Software Development*, Yourdon Press, 1984.

Kroenke, David, *Database Processing: Fundamentals, Design, Implementation* Science Research Associates, Inc., 1977.

Lutton, Bill and Matson, Jo Anna, *Enforcement Management Information System User's Manual*, National Marine Fishery Service, Washington, D.C.

Mitchell, Thaddeus R. and Bell, Robert S. *Drug Interdiction Operations by the Coast Guard: Database and Analysis*, Center for Naval Analysis, Report CRC 413, December 1979.

Page-Jones, Meilir, *The Practical Guide to Structured Systems Design*, Yourdon Press, 1980.

Prague, Cary N. and Hammitt, James E., *Programming with R:base 5000*, Tab Books, Inc., 1986.

ReQuest System User Manual, System Automation Corporation, April 1984.

R:base 4000 User's Manual and Tutorial, MicroRim, Inc., 1984.

R:base 5000 User's Manual, MicroRim, Inc., 1985.

Simpson, Alan, *Understanding R:base 5000*, SYBEX, 1985.

Yourdon, Edward, *Managing the Structured Techniques*, Yourdon Press, 1976.

INITIAL DISTRIBUTION LIST

No. Copies

1. Defense Technical Information Center 2
Cameron Station
Alexandria, VA 22304-6145

2. Library, Code 0142 2
Naval Postgraduate School
Monterey, CA 93943-5002

3. Curricular Office, Code 37 1
Department of Computer Technology
Naval Postgraduate School
Monterey, CA 93943-5000

4. Barry A. Frew, Code 54Fw 2
Department of Administrative Sciences
Naval Postgraduate School
Monterey, CA 93943-5000

5. Vincent Y. Lum 1
Department of Computer Science
Naval Postgraduate School
Monterey, CA 93943-5000

6. Chief of Naval Operations 2
Director, Information Systems (OP-945)
Navy Department
Washington, D. C. 20350-2000

7. RADM W. Merlin 2
Commandant (G-T)
U.S. Coast Guard
2100 Second Street SW
Washington, D. C. 20590-0000

8. Commandant (G-TPP/HRM) 2
U.S. Coast Guard
2100 Second Street SW
Washington, D. C. 20593-0000

9. Commandant (G-PTE-1) 2
U.S. Coast Guard
2100 Second Street SW
Washington, D. C. 20590-0000

10. Commander 2
Coast Guard Group Monterey
100 Lighthouse Avenue
Monterey, CA 93940-1497

11. LT Jon D. Allen 6
U.S. Coast Guard Station Alexandria
Telegraph Road
Alexandria, Virginia 22310-0000

12. LT Heidi R. Lang 6
100 Conant Drive
Buffalo, New York 14223-0000

13. Mr. and Mrs. Gordon G. Allen 1
1555 Dorset Street
South Burlington, VT 05402-0000

14. Mr. and Mrs. Edward P. Lang 1
100 Conant Drive
Buffalo, New York 14223-0000